



---

***SISTEMA DE INFORMAÇÃO GEOGRÁFICA  
APLICADO AO TRANSPORTE DE  
RECLUSOS***

---

Maria Luísa Horta Correia Ramirez Delgado

---

Trabalho de Projecto apresentado como requisito parcial  
para obtenção do grau de Mestre em Ciência e Sistemas  
de Informação Geográfica

---

Instituto Superior de Estatística e Gestão de Informação  
da Universidade Nova de Lisboa

**SISTEMA DE INFORMAÇÃO GEOGRÁFICA APLICADO AO TRANSPORTE  
DE RECLUSOS**

Trabalho de projecto orientado por  
Professor Doutor Pedro da Costa Brito Cabral

Novembro 2011

## AGRADECIMENTOS

Agradeço ao meu orientador, Doutor Pedro Cabral, a clara orientação, todo o seu apoio e pronta disponibilidade.

Agradeço ao Dr. Carlos Borralho, da DGSP, a possibilidade dada para a realização deste trabalho, na área de transportes desse organismo, assim como, às colaboradoras do centro de remoções do GISP, a sua colaboração.

Agradeço ao Jorge e Francisco todo o seu apoio.

# **SISTEMA DE INFORMAÇÃO GEOGRÁFICA APLICADO AO TRANSPORTE DE RECLUSOS**

## **RESUMO**

Os Sistemas de Informação Geográfica (SIG) desenvolveram nos últimos anos funcionalidades específicas para as áreas de transporte, logística e distribuição, que juntam a informação de localização e topológica a algoritmos de optimização em redes, tomando estes softwares o nome de SIG para Transportes (SIG-T).

Este projecto explora as potencialidades de criar ferramentas SIG-T para auxiliar a gestão da frota da Direcção Geral dos Serviços Prisionais (DGSP) no serviço de transporte de reclusos.

Para o efeito, foi construído algum software com o intuito de, a partir de um pedido de transporte, que vai originar uma sequência de viagens, atribuídas a vários transportadores, gerar a sequência de pedidos atribuídos aos vários transportadores em questão.

Foi construída uma rede que reflecte os vários eixos de transporte da DGSP e as suas características. Embora o maior enfoque do projecto seja dado ao tratamento dos pedidos de transporte, também foi explorado o planeamento de viagens a partir de pedidos, usando a mesma rede.

Foram usadas funcionalidades de *routing* e de *Vehicle Routing Problem*, de um software com capacidades SIG-T, a extensão Network Analyst, do ArcGIS, foi construído um *script* em linguagem Python, e este processo ainda foi complementado com procedimentos Oracle.

# **GEOGRAPHIC INFORMATION SYSTEM APPLIED TO THE TRANSPORTATION OF PRISONERS**

## **ABSTRACT**

In recent years Geographic Information Systems (GIS) developed specific functionality in the areas of transportation, logistics and distribution, combining the location and the topological information, to optimization algorithms in networks, taking the name of GIS for Transportation (GIS -T).

This project explores the potential of creating tools to assist GIS-T fleet management of the Direcção Geral dos Serviços Prisionais (DGSP) in the transportation of prisoners.

To achieve this goal, some tools were built with the intention of starting from a transfer request, which will lead to a sequence of trips assigned to multiple carriers, and generate the sequence of orders assigned to the various carriers in question.

A network that reflects the various transport axes and their characteristics was built. Although the main focus of the project is given to transfer requests, a tool that schedules trips using the same network was also built.

Routing and Vehicle Routing Problem tools were used. A GIS-T software, the Network Analyst extension of ArcGIS was used. Python's *script* were developed to solve the addressed problem complemented with some Oracle procedures.

## PALAVRAS-CHAVE

Problema de Roteamento de Veículos

*Routing*

Sistema de Informação Geográfica

Sistema de Informação Geográfica para Transportes

## KEYWORDS

Vehicle Routing Problem

Routing

Geographic Information Systems

Geographic Information Systems for Transportation

## ACRÓNIMOS

**DGSP** – Direcção Geral dos Serviços Prisionais  
**EP** – Estabelecimento Prisional  
**GISP** – Grupo de Intervenção dos Serviços Prisionais  
**IDLE** - *Integrated Development Environment*  
**IO** - Investigação Operacional  
**OR/MS** – *Operation Research and Management Systems*  
**PCV** – Problema do Caixeiro Viajante  
**PRV** – Problema de Roteamento de Veículos  
**SIG** – Sistema de Informação Geográfica  
**SIG-T** – Sistema de Informação Geográfica para Transportes  
**SIP**- Sistema de Informação Prisional  
**SIP-Frota** - Sistema de Informação Prisional - Frota  
**SIT** - Sistema de Informação de Transporte  
**SQL** - *Structured Query Language*  
**TN**- Transportador Nacional  
**TSP**- *Traveling Salesman Problem*  
**VRP**- *Vehicle Routing Problem*  
**XML** - *Extensible Markup Language*

## ÍNDICE DO TEXTO

AGRADECIMENTOS.....	iii
RESUMO.....	iv
ABSTRACT.....	v
PALAVRAS-CHAVE.....	vi
KEYORDS.....	vi
ACRÓNIMOS .....	vii
ÍNDICE DE TABELAS .....	x
ÍNDICE DE FIGURAS .....	xi
1. INTRODUÇÃO .....	1
1.1 Enquadramento .....	1
1.2 Objectivos .....	4
1.3 Pressupostos e Hipóteses .....	5
1.4 Estrutura da Tese .....	6
2. TRANSPORTE E SISTEMAS DE INFORMAÇÃO GEOGRÁFICA .....	7
2.1 Objectivos do capítulo .....	7
2.2 Problemas Clássicos de Transporte .....	7
2.2.1 A Logística e a Investigação Operacional .....	7
2.2.1.1 Logística – Breve História.....	10
2.2.1.2 Investigação Operacional – Breve história .....	12
2.2.2 PRV – Família de Problemas.....	13
2.2.3 Redes.....	16
2.3 Sistemas de Informação Geográfica Aplicados a Transporte –SIG-T.....	18
2.4 Conclusões .....	22
3. PLANEAMENTO DE TRANSPORTE DE RECLUSOS .....	24
3.1 Modelação .....	24
3.1.1 Enquadramento .....	24
3.1.2 Objectivos .....	26
3.1.2.1 Geração de Pedidos de Transporte.....	27



3.1.2.2 Geração de Planeamento de Viagens .....	29
3.1.3 Construção de uma rede de Eixos .....	30
3.1.4 Modelação Geral do Sistema .....	44
3.1.5 Obtenção de Dados .....	45
3.2 Desenvolvimento.....	45
3.2.1 Introdução .....	45
3.2.2 Softwares.....	45
3.2.3 Desenvolvimento.....	48
3.2.3.1 Criação de uma Rede .....	49
3.2.3.2 Partição de Pedidos .....	51
3.2.3.3 Procedimentos Oracle .....	53
3.2.3.4 Procedimentos Manuais .....	54
3.2.3.5 Planeamento de Viagens .....	55
3.3 Conclusões.....	61
4. RESULTADOS .....	64
4.1 Preparação dos dados.....	64
4.2 Resultados.....	65
4.2.1 Resultados de ‘Partição Pedidos’.....	65
4.2.1.1 Outras Questões.....	80
4.2.2 Resultados de ‘Planeamento Viagem’.....	81
5. CONCLUSÃO.....	85
5.1 Avaliação do projecto.....	85
5.2 Conclusões.....	87
5.3 Desenvolvimentos Futuros .....	87
REFERÊNCIAS BIBLIOGRÁFICAS.....	88
ANEXOS.....	90
Anexo 1- <i>Scripts</i> em Linguagem Python.....	90
Anexo 2 - Tabelas e Procedimentos Oracle.....	97

## ÍNDICE DE TABELAS

Tabela 1. Problemas NP-completos.....	14
Tabela 2. Materiais usados em VRP .....	56
Tabela 3. Atributos dos Sub- <i>layers</i> do VRP .....	57
Tabela 4. Parâmetros de dois VRP .....	59
Tabela 5. Tabela com Locais de Origem e Destino dos Pedidos .....	65
Tabela 6. Tabela de Pedidos Iniciais .....	66
Tabela 7. Tabela de Resultados .....	67
Tabela 8. Tabela de Resultados pós procedimentos Oracle .....	69
Tabela 9. Tabela de <i>Depots</i> (Depósitos) do VRP .....	81
Tabela 10. Tabela de input de <i>Orders</i> (Tarefas) do VRP .....	82
Tabela 11. Tabela de <i>Order Pairs</i> do VRP .....	82
Tabela 12. Tabela de input de <i>Routes</i> do VRP .....	83
Tabela 13. Tabela de output de <i>Orders</i> do VRP .....	83

## ÍNDICE DE FIGURAS

Figura 1. Ilustração do Problema de Roteamento de Veículos .....	13
Figura 2. Ilustração de Uma rede .....	17
Figura 3. Mapa contendo os vários níveis de eixos de transporte da DGSP .....	25
Figura 4. Eixo_Norte .....	33
Figura 5. Eixo_Sul.....	34
Figura 6. Eixos_Centro .....	35
Figura 7. Eixos_EP_norte .....	36
Figura 8. Eixos que ligam EPAlcoentre e EPMontijo ao EPLisboa.....	37
Figura 9. Eixos que ligam os EP da zona de Lisboa ao TN, e o TN a EPAlcoentre e EPMontijo.....	38
Figura 10. Eixos que ligam EPLisboa aos tribunais de Lisboa.....	39
Figura 11. Eixos que ligam EPAveiro a tribunais de Aveiro e Coimbra. ....	40
Figura 12. Eixos que ligam alguns EP ao TN.....	41
Figura 13. Eixos que ligam os EP da zona de Lisboa a EPAlcoentre e EPMontijo....	42
Figura 14. Eixos que ligam os EP de Lisboa aos tribunais de Lisboa .....	42
Figura 15. Esquema geral das funcionalidades principais .....	44
Figura 16. Esquema de interligação entre os vários <i>softwares</i> usados .....	46
Figura 17. Modelo de construção de uma rede de eixos.....	50
Figura 18. Fluxo de operações da funcionalidade ‘Partição Pedidos’ .....	53
Figura 19. Fluxo Geral da Partição de Pedidos .....	55
Figura 20. Modelo de criação de um VRP layer.....	60
Figura 21. Modelo Geral de Planeamento de Viagens.....	60
Figura 22. Caminho 635: entre EPLeiria e um tribunal da Covilhã.....	70
Figura 23. Caminho 766: entre EPIzeda e um tribunal de Viseu.....	71
Figura 24. Caminho 37: entre EPCoimbra e um tribunal de Lisboa.....	72
Figura 25. Caminho 90: entre EPLinhó e um tribunal de Beja. ....	72
Figura 26. Caminho 852: entre EPBeja e um tribunal de Sintra .....	73
Figura 27. Caminho 327: entre EPPcruz e um tribunal de Faro. ....	74
Figura 28. Caminho 847: entre EPAveiro e um tribunal de Águeda .....	74
Figura 29. Caminho 974: entre EPGuarda e um tribunal de Aveiro.....	75

Figura 30. Caminho 297: entre EPPferreira e um tribunal de Braga .....	76
Figura 31. Caminho 301: entre EPPferreira e um tribunal de Coimbra. ....	76
Figura 32. Caminho 30: entre EPCoimbra e um tribunal de Coimbra. ....	77
Figura 33 Caminho 616: entre EPLeiria e um tribunal de Faro .....	78
Figura 34. Caminho 336: entre EPPcruz e um tribunal de Lisboa. ....	78
Figura 35. Caminho 616: entre EPLeiria e um tribunal de Faro – ‘rede3’ .....	79
Figura 36. Ilustração de um tipo de <i>snap</i> para toda a rede .....	81
Figura 37. Percurso e paragens da <i>route</i> LP21062011.....	84

# 1.INTRODUÇÃO

## 1.1 Enquadramento

Transporte, Logística e Distribuição são áreas complementares cuja importância tem crescido nos últimos anos devido à globalização e consequente aumento de deslocação de mercadorias e pessoas em espaços cada vez mais amplos. Por outro lado a actual necessidade de eficiência na gestão, e na qualidade dos serviços, devido ao aumento do preço dos combustíveis, a uma muito maior concorrência e maior exigência de qualidade pelos cidadãos em geral, levam à necessidade e à procura de formas de gerir os transportes, menos empíricas e mais apoiadas em métodos científicos.

A Gestão de Frotas e a Gestão de Trânsito são duas áreas que usufruem de conhecimentos científicos matemáticos e computacionais em Modelação de Redes. Associadas à Gestão de Frotas surgem tarefas, tais como o estudo da procura, a definição de rotas de distribuição e o escalonamento de entregas, normalmente associadas a mercadorias, mas começando a emergir também ao nível do transporte de pessoas, o chamado ‘Transporte a Pedido’, em que a procura é bem definida e conhecida de antemão e o passageiro é transportado de porta-a-porta. Este é um tipo de transporte que surge para resolver problemas de grupos com necessidades especiais, como idosos, doentes, e outros.

Em termos científicos alguns destes problemas já têm 50 anos, tendo sido definido como *Vehicle Routing Problem* (VRP), por Dantzig e Ramser (1959) o problema de calcular serviços de distribuição a um custo mínimo, apoiado na área matemática de optimização combinatória.

Como em todas as outras, também esta área tem usufruído em grande escala do recurso às tecnologias de informação e comunicação, e devido à natureza essencialmente espacial e contexto geográfico dos problemas referidos, os SIG produziram um enorme incremento de eficiência na Gestão de Transportes, Logística e Distribuição, nomeadamente em Gestão de Frotas, fazendo surgir um tipo específico de SIG, o SIG-T (ou GIS-T, acrónimo na língua inglesa) que acrescenta as vantagens dos SIG à já antiga ciência dos Transportes e Logística, agregando informações várias e pertinentes para uma boa solução, como sejam a exacta localização dos nós da rede,

permitindo realizar análises espaciais, e não menos importante, permitindo visualizar graficamente redes, locais, percursos no espaço e no tempo.

Os SIG-T permitem criar verdadeiras ferramentas de planeamento de transporte, permitindo antecipar situações através de simulações analisáveis e visualizáveis e 'decidir' de um modo sustentado. A Internet e a computação móvel enriqueceram esta área do conhecimento com capacidades de actualização e utilização em qualquer momento e em qualquer lugar, permitindo actualizar o sistema a todo o momento com novas informações associadas, tanto à alteração da 'procura', como à alteração das características dos 'locais' e 'caminhos'.

Este projecto pretende aplicar algum deste conhecimento em Ciência e Tecnologia, ao planeamento de Transporte de Reclusos. Este planeamento é realizado pelo Grupo de Intervenção dos Serviços Prisionais (GISP), da Direcção Geral dos Serviços Prisionais (DGSP), de um modo totalmente empírico, trabalhoso e pouco eficiente, sendo reconhecido pelos próprios, a urgência de modernização.

No contexto do Transporte de Reclusos, a tipologia da procura insere-se no tipo de transporte de passageiros, 'a pedido' ou seja, conhecido antecipadamente, mas sujeito a alterações de última hora, e que neste caso particular inclui basicamente dois tipos de viagens:

1- Transporte de reclusos entre um Estabelecimento Prisional (EP) e um local próximo, como seja um tribunal, um hospital, ou outro Estabelecimento Prisional, a cargo do próprio Estabelecimento Prisional.

2- Transporte entre os Estabelecimentos Prisionais de todo o país, a cargo de um centro de coordenação, com uma frota para o efeito.

Estas duas tipologias de transporte estão interligadas, mas este projecto focou-se essencialmente no transporte referido no ponto dois, pretendendo criar ferramentas computacionais que consigam responder às necessidades, com maior eficiência, eficácia, e qualidade, recorrendo a informação e análises geográficas, algoritmos de optimização e capacidades de visualização.

Nalgumas regiões do país, o tipo de transporte referido no primeiro item poderá

evoluir futuramente para uma tipologia semelhante, deixando de ser resolvido individualmente por cada Estabelecimento Prisional, e passando a ser coordenado centralmente, se existirem meios para isso, no que, a aplicação que irá ser construída terá também um importante papel, ficando patente a sua importância presente e futura.

Algumas das características específicas deste problema:

- Existem dois eixos principais de transporte, cujos nós, funcionam como nós principais, aos quais se ligam os restantes eixos;
- São realizadas viagens semanais ao longo desses eixos;
- Nos nós desses eixos podem sair e entrar passageiros;
- O percurso realizado ao longo dos eixos principais é fixo;
- Existe uma data limite para que os passageiros estejam no seu destino;
- Pretende-se que os recursos utilizados sejam mínimos;
- O problema abarca basicamente todo o país.

O enfoque principal do projecto foi o estudo de como é que os SIG incorporaram os algoritmos de Optimização em Redes aplicados aos Transportes ou dito de outra forma, que vantagens é que os SIG trouxeram ao Problema de Roteamento de Veículos<sup>1</sup> e à aplicação dos algoritmos de optimização em redes aplicados aos Transportes a ele associados. Pretendeu-se compreender os SIG-T e a sua aplicação a problemas concretos, nomeadamente, ao tipo de problema do estudo de caso, para, seguidamente, se construírem ferramentas computacionais adequadas.

Neste projecto foram exploradas as potencialidades da extensão Network Analyst do software ArcGIS, na aplicação das referidas operações de optimização, assim como a associação ArcGIS/Python. O sistema usou recursos da rede informática da DGSP. Pretendeu-se construir algo, expansível no futuro a outros tipos de viagens. Os ganhos serão em tempo gasto e eficiência de planeamento, e na própria informatização de dados que não o estavam.

O trabalho foi desenvolvido para ser aplicado ao trabalho do GISP e usará os

---

<sup>1</sup> Problema de Roteamento de Veículos (PRV) é a tradução usada no Brasil, para *Vehicle Routing Problem (VRP)*. Neste documento será usado o termo usado no Brasil para nos referirmos ao problema, mas quando nos referirmos a uma funcionalidade que resolve o problema, será usada a expressão em língua inglesa.

dados do SIP (Sistema de Informação Prisional) e SIP-Frota (Sistema de Informação Prisional-Frota), sistemas informáticos da DGSP que gerem dados de reclusos e de viaturas.

No contexto deste trabalho chamar-se-á ao GISP, TN (Transportador Nacional), e aos reclusos, transportados, porque se considera que são termos mais adequados a um problema de transporte e distribuição, permitindo compreendê-lo com mais clareza.

Mais do que representar o exacto sistema de transporte da DGSP, pretende-se representar um funcionamento de base, e alguns casos particulares, apresentando uma solução que os resolva os problemas dentro desse sistema.

## **1.2 Objectivos**

Os objectivos deste projecto são:

1. Avaliar os SIG-T na perspectiva da sua aplicabilidade a problemas de Transporte e Distribuição e na sua capacidade de incorporar algoritmos de Optimização em Redes, nomeadamente em relação a:

- PRV - o problema e suas variantes;
- SIG-T - as capacidades de um SIG na área de Transporte e Distribuição;
- SIG-T – a sua aplicação ao problema concreto do estudo de caso do Transporte de Reclusos.

2. Construir uma ferramenta SIG, integrável com o SIP, aplicável ao planeamento de Transporte de Reclusos, que permita:

- Criar automaticamente pedidos para serem usados pelo TN;
- A partir de uma lista de Pedidos de Transporte criar um planeamento de Viagem.



### **1.3 Pressupostos e Hipóteses**

Os pressupostos subjacentes ao desenvolvimento deste projecto são os seguintes:

1. O TN é responsável por transportar reclusos de um EP para outro, ao longo de todo o território português, com uma frequência de cerca de 100 transferências por semana, sendo o processo de planeamento dessas viagens, demorado, difícil e pouco eficiente, pois é realizado sem recurso a qualquer ferramenta informática.
2. Existe conhecimento científico ao nível da optimização combinatória que permite resolver quase todos os problemas de planeamento de transporte e distribuição que visam optimizar a gestão dos recursos.
3. Devido à componente espacial que lhe é inerente, os transportes são umas das áreas de grande aplicabilidade dos SIG, pela sua capacidade única de integração de espaço, tempo, e outra informação com eles relacionada e sua visualização em mapas, fazendo surgir os SIG-T, softwares especializados em armazenar, processar e comunicar informação de transporte.

No desenvolvimento foram formuladas as seguintes hipóteses:

1. É possível utilizar algoritmos de Optimização Combinatória que permitam automatizar e optimizar o problema concreto da criação de pedidos e planeamento de viagens no transporte de reclusos, facilitando o processo e melhorando os resultados.
2. A existência de um software SIG-T que associe um algoritmo apropriado de optimização combinatória à possibilidade de visualização em mapas, permitirá construir ferramentas adequadas a este problema
3. O recurso a visualização geográfica de pedidos e de percursos, proporcionará uma percepção e entendimento dos problemas e das soluções, no espaço e no tempo, que representará por si só um ganho inestimável no processo de tomada de decisão.

## **1.4 Estrutura da Tese**

O capítulo 1 contém uma introdução, referindo o enquadramento, objectivos e pertinência do caso de estudo, assim como a estrutura da tese.

No capítulo 2 faz-se uma revisão bibliográfica sobre a evolução de problemas semelhantes ao do caso de estudo e de como os Sistemas de Informação Geográfica podem ajudar a resolvê-los. Termina com uma conclusão sobre como estes conhecimentos podem ser utilizados na resolução do caso de estudo.

O capítulo 3 inclui a modelação e desenvolvimento da solução: começa por fazer um enquadramento mais detalhado do estudo de caso, especificando bem os objectivos específicos e descrevendo depois toda a modelação e desenvolvimento realizados para os concretizar. Termina com uma conclusão sobre o resultado conseguido, sobre as ferramentas usadas e sobre as facilidades ou dificuldades encontradas.

No capítulo 4 são apresentados e analisados os resultados obtidos através das ferramentas construídas, ao longo de várias fases em que foram sendo encontradas formas de os melhorar, ou seja, não só os resultados, mas também a própria evolução do projecto, em função dos resultados.

O capítulo 5 é o capítulo das conclusões finais, da análise crítica dos resultados obtidos e dos métodos usados, em função dos objectivos iniciais, e inclui propostas de continuação do projecto.

## **2. TRANSPORTE E SISTEMAS DE INFORMAÇÃO GEOGRÁFICA**

### **2.1 Objectivos do Capítulo**

Este capítulo pretende expor a pesquisa realizada sobre a temática do problema e das suas soluções.

O problema de transporte que aqui se pretende estudar pertence à área da distribuição, que por sua vez é parte da logística.

Uma vez que este problema se enquadra num conjunto clássico de problemas de logística, muito anteriores ao surgimento dos Sistemas de Informação Geográfica, considera-se interessante incluir o contexto em que surgiram e como foram recebidos e resolvidos pela sociedade empresarial e científica.

Problemas como o do estudo de caso, são problemas de ordem prática, que sempre existiram ligados às actividades humanas. Para a sua resolução, surgiram ciências empresariais como a logística, que inclui a distribuição e os transportes, e vários ramos da ciência matemática, como a optimização matemática, a optimização combinatória, e a teoria de grafos, assim como, a Investigação Operacional que aplica as últimas às primeiras.

No caso dos problemas de transportes, por um lado, a teoria dos grafos permitiu a sua representação e modelação, de um modo simples e intuitivo, e por outro lado, a resolução de alguns dos seus problemas assumiu tal complexidade que se tornou um dos maiores desafios matemáticos.

Finalmente os SIG vieram dar o seu contributo adequando as suas estruturas à implementação de redes e acoplando os métodos matemáticos.

### **2. 2 Problemas Clássicos de Transporte**

#### **2.2.1– A Logística e a Investigação Operacional**

São muitos, e estão muito interligados os termos e as ciências relacionadas com a área destes problemas alguns dos quais surgiram para os resolver. Considerou-se

importante clarificar cada um deles, evidenciando as suas semelhanças ou diferenças.

Investigação Operacional (IO) – Ramo da matemática, interdisciplinar, que surgiu para resolver problemas de Logística durante a 2ª guerra mundial. Utiliza métodos científicos e modelos matemáticos com recurso a computador, para resolver problemas de apoio à decisão (de gestão, engenharia e economia), estudando a efectiva utilização da tecnologia pelas organizações.

Optimização Matemática – Selecção do melhor elemento de entre um conjunto de alternativas. No caso mais simples trata de maximizar ou minimizar uma função real, escolhendo o melhor de entre vários valores de um conjunto. Inicialmente chamada Programação Linear, criada por George B. Dantzig.

Programação Linear (PL) - Visa minimizar uma função objectivo linear, sujeita a um conjunto de restrições. Esta metodologia tem uma grande variedade de aplicações, em problemas de transporte, que envolvem várias origens e destinos, com o objectivo de otimizar uma solução a um custo mínimo (Rodrigue, 2009).

Programação Linear Inteira - Um problema de programação inteira é um programa de optimização matemática, no qual algumas ou todas as variáveis do problema pertencem ao conjunto dos números inteiros (não contínuos). Isto dificulta a sua resolução, pois não permite usar o método base da programação linear (simplex). Apesar de ter um número finito de soluções possíveis, esse número pode ser muito grande e impossível, portanto, de resolver em tempo útil.

Optimização em Redes – Um caso especial de programação linear, com vantagens em tempo e por serem mais intuitivas. Resolve problemas de fluxo de redes que são considerados suficientemente importantes para que se tenha gerado muita pesquisa em algoritmos especializados para suas soluções.

Simplex - Algoritmo para resolver problemas de programação linear, utilizado em cerca de 85% das maiores empresas.

Optimização Combinatória – ramo científico da Matemática Aplicada (subconjunto da Optimização Matemática) e da Ciência da Computação, que estuda problemas de optimização em conjuntos finitos, e que surgiu para resolver problemas de Investigação Operacional, nomeadamente o Problema de Roteamento de Veículos e o Problema do Caixeiro-viajante.

Análise de Redes – Área de tecnologia da informação e das ciências sociais que trata do processo de analisar qualquer tipo de rede por meio da teoria das redes. A análise inclui descrições da *estrutura e otimização*, como a Análise de Caminho Crítico e propriedades como atribuição de fluxo.

Problema NP-Completo – problemas para os quais não se conhece uma solução determinística capaz de ser executada em tempo polinomial.

Heurísticas – Recorrem a ‘truques’ ou ‘atalhos’ para chegar à melhor solução, sem ter que avaliar todas elas, e baseiam-se em conhecimento relativo ao problema que está a ser resolvido, adquirido através da experiência ou da intuição. Digamos que o conhecimento empírico vem colaborar com o conhecimento científico. Produzem resultados não cientificamente comprováveis mas são computacionalmente eficientes.

Meta-heurísticas - Para além das heurísticas que são métodos muito específicos para determinado problema, existem ainda as meta-heurísticas, métodos mais genéricos que servem a qualquer problema.

Grafos – Uma representação visual de um determinado conjunto de dados e da ligação existente entre alguns dos elementos desse conjunto. (Wikipedia, 2011). Um grafo constitui o modelo matemático ideal para o estudo das relações entre objectos discretos de qualquer tipo.

Teoria dos Grafos – A base matemática para representação de redes (particularmente para modelação e análise) (Miller, 2001).

Rede - Um grafo cujos vértices e/ou arestas têm associado valores numéricos (custos, capacidades e/ou oferta e procura). A terminologia utilizada em redes é nós (nodos) e arcos, em vez de vértices e arestas, respectivamente. De uma maneira geral, os conceitos utilizados para grafos são extensíveis a redes. (Barrico, 1998).

Árvore – Uma árvore é um grafo com um e um só caminho entre quaisquer dois vértices.

Logística - Área da gestão responsável por prover recursos, equipamentos e informações para a execução de todas as actividades de uma empresa. (Wikipedia, 2011).

Routing – Processo de procura de um caminho dentro de uma rede. É usado em vários tipos de redes, incluindo redes telefónicas, redes de dados electrónicos (como por exemplo a internet), redes de transporte.

Transporte - Existe para superar discrepâncias geográficas em recursos, bens e serviços, movimentando materiais, pessoas ou informações, entre onde as coisas estão, e onde as coisas são necessárias (ou vice-versa) (Miller, 2001).

Sistema de Transporte - Pode ser definido como o conjunto de relações entre nós, redes e procura, relativa a movimentos de pessoas, mercadorias, e informação. Nós, são os locais de origem, destino e mudanças de movimento (Rodrigue, 2009).

Geografia de Transportes - A maioria dos modelos aplicados à geografia dos transportes são derivados de métodos matemáticos. A Investigação Operacional contribuiu consideravelmente para esta área, oferecendo um conjunto de métodos para otimizar a distribuição e programação de recursos de transporte (Rodrigue, 2009).

Os métodos mais utilizados em geografia de transporte incluem, análise de redes, também conhecida como teoria dos grafos, que é usado para estudar forma e estrutura de redes de transporte (Rodrigue, 2009).

Impedância – peso atribuído aos eixos ou nós de uma rede, e no qual é baseada a otimização dos percursos.

### **2.2.1.1 Logística - Breve História**

‘Desde os tempos bíblicos, os líderes militares já se utilizavam da logística. As guerras eram longas e geralmente distantes e eram necessários grandes e constantes deslocamentos de recursos. Para transportar as tropas, armamentos e carros de guerra pesados aos locais de combate eram necessários planejamento, organização e execução de tarefas logísticas, que envolviam a definição de uma rota; nem sempre a mais curta, pois era necessário ter uma fonte de água potável próxima, transporte, armazenagem e distribuição de equipamentos e suprimentos.

Na antiga Grécia, Roma e no Império Bizantino, os militares com o título de *Logistikas* eram os responsáveis por garantir recursos e suprimentos para a guerra.’ (Wikipedia, 2011).

É a Antoine-Henri Jomini, ou Jomini, contemporâneo de Clausewitz, que se deve, pela primeira vez, o uso da palavra "logística", definindo-a como "a acção que conduz à preparação e sustentação das campanhas.

A logística como ciência teve sua origem nas teorias criadas e desenvolvidas

pelo Tenente-Coronel Thorpe, do Corpo de Fuzileiros Navais dos Estados Unidos da América que, no ano de 1917, publicou o livro "Logística Pura: a ciência da preparação para a guerra". É interessante referir que O Almirante Henry Eccles, considerado como o "pai da logística moderna" (Wikipedia, 2011), em 1945, ao encontrar a obra de Thorpe, comentou que, se os EUA seguissem seus ensinamentos teriam economizado milhões de dólares na condução da 2ª Guerra Mundial.

Como vimos, a Logística é uma ciência muito antiga, sempre presente na organização militar e nas guerras, ligada ao objectivo de sobrevivência das pessoas, mas também no comércio. Desde o pós-guerra é parte integrante da complexa organização empresarial actual, associada ao ganho de valor pela empresa com o objectivo de obter vantagens competitivas.

Em (Neto, 2002), refere-se que o termo Logística, de acordo com o Dicionário Aurélio, vem do francês *logistique* e tem como uma de suas definições ‘a parte da arte da guerra que trata do planeamento e da realização de: projecto e desenvolvimento, obtenção, armazenamento, transporte, distribuição, reparação, manutenção e evacuação de material (para fins operativos ou administrativos)’.

Mas é importante notar que no dicionário de termos do Council of Supply Chain Management Professionals, a definição de logística como, ‘o processo de planeamento, implementação e controlo de procedimentos para o eficiente e eficaz transporte e armazenamento de mercadorias, incluindo os serviços e informações relacionadas, desde o ponto de origem até o ponto de consumo com o propósito de conformidade com os requisitos do cliente (Vitasek, 2010), inclui também as informações na lista de tarefas da Logística.

Portanto, o transporte é uma das tarefas da logística. Por outro lado, transportar mercadorias, garantindo a integridade da carga, no prazo combinado e a baixo custo exige o que se chama "logística de transporte".

Como vimos a logística existe desde sempre associada a questões de sobrevivência, muitas vezes, mas não só, associada a questões militares, e a partir dos anos 60 do século passado, passou também, a fazer parte da sobrevivência das empresas.

Isto deveu-se ao grande avanço tecnológico, posterior à segunda guerra mundial, que alterou totalmente o tipo de vida, dos indivíduos, dos negócios e das organizações,

dos países da Europa e da América, gerando um enorme aumento do consumo e da produção por um lado, da informação e da comunicação por outro. Paralelamente a estes, também o desenvolvimento dos meios de transporte e as vias de comunicação alteraram totalmente as noções de tempo e de espaço.

Seja pela grande concorrência que se foi estabelecendo, como pela própria dificuldade em gerir tão grandes quantidades de produtos e serviços envolvidos, num espaço e num tempo muito grandes e muitos preenchidos, que a partir dos anos 80, tanto os novos como os antigos problemas das empresas, ao nível da logística, revestiram-se de tal complexidade que passaram a exigir cálculos e análises realizadas por procedimentos muito eficientes e máquinas muito sofisticadas.

Após os anos 80, a logística passa a ter realmente um desenvolvimento revolucionário, devida à globalização, pela alteração da economia mundial e pelo grande uso de computadores na administração.

Viu-se como a logística passou de uma tarefa muito prática realizada de um modo empírico e recorrendo ao engenho humano, a uma ciência dotada de regras sistemáticas, métodos e modelos matemáticos, tanto podendo estar ligada a meros objectivos de sobrevivência como a objectivos de rentabilização de recursos.

#### **2.2.1.2 Investigação Operacional - Breve História**

Como uma disciplina formal, a Investigação Operacional surgiu na 2<sup>a</sup> Grande Guerra e foi definida como ‘um método científico de prover os departamentos executivos com uma base quantitativa para a tomada de decisão’.

Vários cientistas do Reino Unido e dos Estados Unidos da América, dos quais apenas se refere Dantzig, estudam meios para melhorar as tomadas de decisão nas áreas de logística.

Para o efeito surgiu o ramo matemático Programação Linear para controlar despesas e reduzir custos e aumentar as perdas do inimigo, tendo sido mantido secreto até 1947, ano em que Dantzig publicou o método *simplex*.

O exemplo original de Dantzig de calcular a melhor atribuição de 70 pessoas a 70 tarefas, exemplifica a enorme utilidade da Programação Linear. O poder computacional requerido para testar todas as permutações e escolher a melhor é bastante



grande, uma vez que o número de possíveis configurações excede o número de partículas do universo. No entanto, basta um único momento para encontrar a melhor solução se o problema for posto como um problema de investigação operacional ao qual se aplique o método *simplex*.

Depois da guerra esses conhecimentos passaram a ser usados no dia-a dia das empresas e pela indústria.

### 2.2.2 PRV – Família de Problemas

Como já foi referido dois dos mais estudados problemas de otimização combinatória são o Problema do Caixeiro-viajante (PCV) e o Problema de Roteamento de Veículos (PRV), sendo normalmente estudados em conjunto. O PCV é mais antigo, podendo o PRV ser visto como uma extensão do primeiro.

O PVC surgiu como um problema do dia-a-dia dos caixeiros-viajantes, que pretendiam escolher o modo mais eficiente de realizar a sua tarefa, e consiste em calcular a partir de um conjunto de locais o menor circuito que visite todos os locais, passando apenas uma vez em cada um deles.

Mais complicado o PRV, pretende calcular a mais eficiente solução para que uma frota de veículo distribua mercadoria por vários clientes, voltando ao ponto de partida. Neste novo problema, mantém-se o problema dos caminhos, mas agora para uma frota, ou seja, vários veículos, como representado na figura 1.

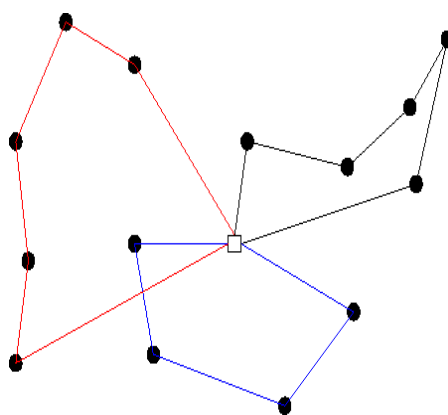


Figura 1. Representação do Problema de Roteamento de Veículos

Fonte: (Wikipedia, 2011)

Ambos pertencem à classe dos problemas NP-completos, cuja complexidade é

mostrada na tabela 1. Por maior que seja a capacidade computacional de um computador (nº de rotas por segundo), sendo a quantidade de rotas a analisar,  $(n-1)!$ , um número muito grande, o tempo necessário a essa análise torna a sua resolução inviável, á medida que o nº de locais ( $n$ ) aumenta.

$n$	rotas por segundo	$(n - 1)!$	cálculo total
5	250 milhões	24	insignificante
10	110 milhões	362.880	0.003 seg
15	71 milhões	87 bilhões	20 minutos
20	53 milhões	$1,2 \times 10^{17}$	73 anos
25	42 milhões	$6,2 \times 10^{23}$	470 milhões de anos

Tabela 1. Problemas NP-completos

Fonte: (Wikipedia, 2011)

Enquanto o PCV diz respeito a *routing*, o PRV inclui *routing*, e *sheduling*, podendo o segundo ser considerado uma generalização do primeiro, ou o primeiro um caso particular do segundo.

Eles e todas as suas variantes constituem uma família de problemas que conseguem reflectir uma imensidade de situações da vida real que se podem ‘modelar’ a partir deles.

A sua aplicação é cada vez mais solicitada Por serem tão concretos, e no entanto tão difíceis de resolver, suscitam bastante curiosidade e são um desafio (fascinante) para os matemáticos. ’

Segundo dados da Wikipédia, apesar de apenas em 1930 ter sido formulado como um problema matemático, a sua origem remonta 100 anos atrás, referido num manual de um caixeiro viajante, e em 1857, formulado pelo matemático Hamilton. Em 1930, é Karl Menger define a forma geral do problema, assim como a sua solução. Em 1950 é introduzido por Hassler Whitney, o nome *Travelling Salesman Problem*.

Em relação ao PRV, foi formulado na literatura científica, por Dantzig e Ramser em 1959, (Dantzig, 1959) na resolução de um problema de distribuição de gasolina por várias estações de serviço.

Desde a sua formulação por Dantzig e Ramser, seguida de uma heurística criada

por Clarke e Wright que melhorou a formulação de Dantzig-Ramser (Toth, 2002), muitos modelos e algoritmos exactos ou recorrendo a heurísticas têm sido desenvolvidos.

Por se ter tornado, nos últimos 50 anos, um desafio para os estudiosos, a sua formulação na literatura científica e a sua exploração até à exaustão, e de todas as suas variantes, é imensa.

Olhando o problema mais de perto, dados um conjunto de clientes, e um conjunto de veículos:

As características típicas dos clientes são (Toth, 2002):

- Local em que se encontram;
- Quantidade de mercadorias;
- Períodos do dia em que estão disponíveis;
- Tempo de demora no cliente.

As características típicas dos veículos são (Toth, 2002):

- Local do depósito inicial e eventualmente final;
- A sua capacidade;
- Possíveis subdivisões dentro do veículo;
- Dispositivos para carregar e descarregar mercadorias;
- Subconjunto de ruas que podem ser atravessadas;
- Custos associados à sua utilização.

O objectivo é criar um conjunto de percursos, um para cada veículo, que permitam realizar todas as tarefas, minimizando um determinado valor, que pode ser o custo total, o tempo gasto total, as distâncias totais percorridas, o número de veículos ou combinações, ou outros gerados através de combinações destes.

A grande diversidade de situações, gera problemas muito complexos, sendo o estudo das suas soluções, focado separadamente, nas suas muitas e já referidas variantes:

PRVC – PRV com capacidade máxima, com capacidade uniforme: a quantidade de cada *route* não pode ultrapassar a capacidade do veículo, apenas entregas;

PRVJT- PRV com janelas de tempo: cada entrega tem que ser feita dentro de um limite de tempo;

PRVP – PRV periódico extensível a um período maior do que um dia;

PRVMD – PRV com múltiplos depósitos (se os clientes estão agrupados à volta dos depósitos então este problema dará origem a vários PRV simples; senão, surge o PRVMD que exige a atribuição de veículos a depósitos; cada veículo começa e acaba no mesmo depósito);

PRVER - PRV com entregas e recolhas nos clientes (tem a particularidade de para uma mesma mercadoria, a recolha tem que anteceder a entrega);

Todas estas variantes podem tornar-se mais e mais detalhadas e originar novas variantes.

O caso particular de um único veículo e sem outras restrições operacionais, é o PCV (Toth, 2002).

Em (Toth, 2002) o autor lembra que o PRV não é apenas usado em entrega ou recolha de mercadorias, sendo útil em várias outras situações da vida real também relacionadas com transportes, como sejam aplicações de recolha de lixo, limpeza de ruas, autocarros escolar, serviços ‘dial-a-ride’ e transporte de pessoas com deficiência. Segundo (Toth, 2002) muitas outras aplicações interessantes de PRV são referidas na literatura sobre Investigação Operacional.

Que por um lado, devido aos avanços na área de algoritmos de *routing*, por outro, ao desenvolvimento de novos softwares e tecnologias de informática, o PRV é uma das grandes histórias de sucesso da Investigação Operacional (Laporte, 1995), é referido por Laporte na sua bibliografia, contendo 500 referências, relativas a quatro problemas clássicos de transporte entre os quais os dois problemas aqui referidos.

### **2.2.3 Redes**

Os problemas acabados de referir são resolvidos no âmbito da investigação operacional, através de optimização combinatória, e para além disso pertencem ao grupo de problemas, que por lidarem com objectos discretos, podem ser modelados através de redes.

Redes são estruturas matemáticas que representam os problemas através de relações entre os seus elementos, ou seja, entre os objectos discretos que os definem.

Graficamente esses objectos são representados através de pontos e as relações entre eles são representados através de linhas que os ligam, tal como representado na figura 2.

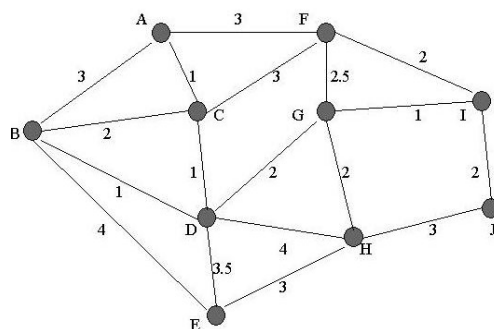


Figura 2. Ilustração de uma rede

Fonte: (Lorena, 2003)

A análise de redes dedica-se ao estudo da estrutura das redes e das operações de optimização aplicáveis a essas estruturas. Em (Lorena, 2003), o autor refere que as redes são entidades formadas por pontos (*nós ou vértices*) e linhas (*arcos ou arestas*) que descrevem de maneira natural vias públicas, conexões de água, telefonia, e outros. Segundo (Miller, 2001) arcos ligam nós, podendo representar ligações físicas (estradas) ou relações lógicas (serviços aéreos entre duas cidades).

Redes são portanto o modelo por excelência para representar problemas de transporte, podendo os arcos eventualmente representar ligações lógicas, entre nós que representem locais físicos.

É usado o conceito espacial de distância combinado com outras informações, assim como um importante conceito para a modelagem de problemas em redes que é o conceito de caminho mínimo, entre dois pontos na rede (Lorena, 2003).

Na figura acima, os valores associados aos arcos são usados como impedância no cálculo do caminho mínimo. Para além da distância espacial pode ser usado, o ‘tempo’ ou outra informação, como por exemplo o ‘nº de veículos’.

As redes actuais de transporte são bastante complexas, assim como a resolução dos seus problemas. Segundo (Miller, 2001) as redes de transporte lidam sempre com arcos direccionados, em que está definido o sentido do fluxo.

### 2.3 Sistemas de Informação Geográfica Aplicados a Transporte – SIG-T

Este capítulo pretende descobrir o que é um SIG-T, e perceber porque é que a área dos Transportes deu origem a um SIG específico.

Tentando perceber quando e porquê surgiram os SIG-T, não se encontra uma referência a uma primeira utilização da expressão Sistemas de Informação Geográfica para Transportes (SIG-T). Este acrónimo surgiu por se ter tornado usual, nos últimos anos, a utilização de SIG em aplicações de transportes, passando a ser usado em conferências sobre SIG, dando origem a conferências específicas sobre o tema como o GIS-T Symposium, discutido em Conferências sobre Transportes como o *Transportation Research Forum*, referido em Publicações sobre Transportes e até publicações específicas de SIG-T, como o *IVHS Journal*, publicada desde 1993 (Waters, 1999), passando a ser tema de estudos e livros, e finalmente originando o desenvolvimento de pacotes de SIG, especificamente para SIG-T.

Em (Miller, 2001) e (Waters, 2005) é referido que em 1993, Maguire escreveu, que, sem sombra de dúvida, era possível afirmar, de forma inequívoca, que o SIG-T ‘chegou’. Em (Thill, 2000) o autor afirma que os fins dos anos 80 viram o uso generalizado de Sistemas de Informação Geográfica na área dos transportes, representando mesmo uma das mais importantes aplicações da tecnologia SIG. O acrónimo SIG-T é muitas vezes usado para referir aplicações e adaptações de SIG às áreas de pesquisa, planeamento, e gestão em transportes (Thill, 2000).

No entanto, será interessante analisar as razões que estão na base de uma explosão tão tardia, notando o facto interessante referido por vários autores de que alguns dos pioneiros do SIG pertenciam à área dos transportes. Em (Goodchild, 2000) o autor lembra que o ímpeto inicial dos SIG veio de um grupo de estudantes de Geografia Quantitativa da Universidade de Washington em 1950, entre os quais se encontrava um aluno chamado Duane Marble, que mais tarde viria a desenvolver uma forma rudimentar de um SIG. Também (Thill, 2000) refere que os pioneiros em SIG na Univ. of Washington e Northwestern University foram de facto cientistas dos transportes.

Assim, é interessante verificar que a área dos transportes impulsionou inicialmente o desenvolvimento de um SIG, depois seguiu outros caminhos e acabou

por regressar às origens.

Perante estes factos põem-se duas questões: 1) Porquê, a adesão tardia, e 2) Porquê, apesar de tarde. Estas duas questões no fundo referem o que os SIG têm de atraente para a área dos transportes e o que lhes faltava e tiveram que incorporar para tornar viável essa utilização.

Tal como referido em (Rodrigue, 2009), o enfoque das pesquisas em SIG-T segue duas direcções diferentes mas complementares. Este autor refere que, enquanto alguma investigação se foca em questões de, como pode o SIG ser melhorado para atender as necessidades das aplicações de transporte, outras investigações focam-se em, como pode um SIG facilitar e melhorar os estudos relativos a transportes.

O que é certo, é que o inicial afastamento do SIG, e o recurso ao uso de sistemas de informação (não geográficos) específicos para transportes leva a deduzir que existe suficiente complexidade e especificidade na área dos transportes em geral, para originar um tipo de sistema de informação que se distingue dos outros, e que essa complexidade não sendo totalmente suportada pelos SIG iniciais, era muito mais valiosa do que a mais-valia ganha pelo uso de um SIG dos anos 70.

Na realidade, como referido em (Thill, 2000) as primeiras aplicações de SIG eram poucas e incapazes de ultrapassar as limitações da tecnologia, na manipulação de dados de transporte, associados a complexos modelos de redes, e não foi feito um esforço para o melhorar devido a uma topologia ainda muito ‘bruta’ (Thill, 2000).

Esta pesquisa deverá obrigatoriamente incluir algum conhecimento sobre um ITS, suas funcionalidades, modelo e limitações, para melhor compreender o afastamento inicial, e bem compreender a força da letra ‘T’.

Segundo (Keenan, 2008), foi no âmbito da área de pesquisa em OR/MS, e portanto num contexto de resolução de problemas do dia-a-dia das empresas, que surgiu o interesse em juntar SIG e técnicas de *routing*. Isto, devido à necessidade de criar modelos cada vez mais realistas, ou seja, mais representativos dos problemas práticos.

Segundo o referido autor, a realidade dos problemas, de entrega e recolha varia consoante o local, regional ou urbano, diminuindo ou aumentando a escala do detalhe, pelo que técnicas de análise de redes baseadas em SIG., seriam de grande utilidade.

Também, segundo (Lorena, 1999), sendo os SIG poderosas ferramentas de análise espacial e o seu comércio, um dos segmentos que mais crescem nos Estados Unidos,

torna-se crítica a integração de técnicas e filosofia da Pesquisa Operacional nesta tecnologia, sabendo que os problemas de Optimização Combinatória aparecem combinados à análise de redes de fluxo de dados (estradas, ruas, canais de comunicação, e outras), representados por segmentos de linha, em que os algoritmos clássicos de optimização de fluxo em redes são directamente aplicáveis. Problemas mais complexos de localização de centros (escolas, ambulatórios, e outros) e roteamento de veículos podem também ser tratados levando-se em conta várias informações espaciais.

Segundo (Thill, 2000) um SIG, e portanto também um SIG-T, diferencia-se dos outros sistemas pela sua complexidade funcional, juntando geo-visualização + database management features + capacidade de análise. Também, segundo (Thill, 2000) a informação gerida por um SIG é a mesma que gerida por um sistema 'normal' só que referenciada de um modo diferente, como por exemplo, no caso de um acidente, em que o outro sistema o associaria a um índice ou a uma data e uma matrícula, e no caso de um SIG ele é referenciado através de uma localização sobre a superfície da terra.

Também em (Filho, 2007) é referido que a vantagem em usar um SIG está na habilidade de associar a cada arco e nó da rede um conjunto de atributos que podem ser manipulados e visualizados graficamente. Além disto, alguns algoritmos bem conhecidos, como por exemplo para minimizar a distância e o tempo percorridos em uma rota ou determinar a rota de menor custo entre dois pontos de interesse, estão imediatamente disponíveis para serem integrados nesses sistemas.

Segundo (Silva, 2003), redes e matrizes são essenciais na solução da maioria de problemas de transporte.

Voltando ao trabalho de (Rodrigue, 2009), a análise da aplicação de SIG a transportes, é realizada segundo 3 categorias: Modelo de Dados, Análises e Aplicações. É referido, que de entre os 2 modelos de dados tradicionais em SIG, a análise de redes baseada na teoria dos grafos que tipicamente representa uma rede como sendo um conjunto de nós ligados por um conjunto de arestas tornará o modelo de dados vectorial, melhor candidato para aplicações em transportes. No entanto, é também referida a importância de um modelo de dados que suporte referência linear, para além dos modelos típicos de um SIG, e são enaltecidas as vantagens desta referência unidimensional, e referido o desenvolvimento de 'dynamic segmentation' data model expressamente para a comunidade SIG-T.



Também, segundo (Thill, 2000) no contexto dos transportes, três classes de modelos SIG são relevantes: campos, modelos discretos e redes, mas de facto muitas aplicações apenas usam o modelo de rede, sendo essa é uma das características de pesquisas em GIS-T: exclusão dos outros modelos. Em (Thill, 2000), o autor considera o modelo de rede elegantemente simples e funcional, com a sua estrutura de arcos/nós que representa através de uma dimensão um objecto que necessitaria de 2 ou 3 dimensões para ser representado na superfície da terra.

Em (Goodchild, 2000) é referido que um dos tipos mais facilmente identificados pelo olho humano são as linhas de transporte - rios, canais, caminhos-de-ferro, e estradas - mas lembra que aqui, o simples modelo linear também pode ter problemas, pois os rios desaguam em lagos e no mar, e pessoas e carros podem sair da estrada e ir para estacionamento. Apesar disso, rios e estradas são elementos relativamente estáveis, sendo também o sistema linear um bom modo de referenciar bastantes actividades humanas.

Mais uma vez, é referido como o sistema linear permite referenciar objectos em um espaço um-dimensional com um simples número, sendo esses referenciais muito mais simples do que os de duas ou três, pois apenas uma medida é necessária, sendo a distância mais fácil de calcular ou estimar intuitivamente, principalmente entre pontos próximos. Por isso, as sociedades humanas sempre optaram por identificar locais através de endereços de estrada e usar este sistema na distribuição de correio em detrimento do sistema bi-dimensional de latitude e longitude.

Não menos importante é a facilidade de armazenar dados um-dimensionais em computador.

Em relação às mais-valias típicas dos SIG, e uma vez que o que existe dentro das organizações ao nível das aplicações de transportes não será um Sistema de Informação de Transporte (SIT), mas sim vários ou seja várias aplicações não integradas, constitui um sério desafio para um SIG-T agregar informação dispersa e multi-temática, o que aliás é vocação de qualquer SIG, a integração natural através do atributo comum 'local'.

Por outro lado, a área dos Transportes tendo-se tornado muito interdisciplinar, funciona ela própria como um elemento agregador, para o qual o SIG se torna uma ferramenta privilegiada.

Em resumo, um SIG-T diferencia-se essencialmente de um SIG, pelas suas

capacidades de modelação de redes linearmente referenciadas e pelas suas capacidades de análise ao nível dessa mesma rede.

As grandes vantagens de um SIG-T são em resumo: Integração, Visualização e Análise de Redes.

Segundo (Miller, 2001), analistas de transporte e decisores, usam-no como ferramenta em planeamento, projecto e gestão de infra-estruturas, planeamento e operacionalização de trânsito, análise e controlo de tráfego, análise de segurança, avaliação de impacto ambiental, prevenção do risco, configuração e gestão de complexos sistemas logísticos.

## **2.4 Conclusões**

Através desta revisão bibliográfica, torna-se patente o quanto a ciência se tem dedicado nos últimos anos, e com bons resultados, a problemas de procura de caminhos através de redes, e o quanto eles são usados e aplicados aos negócios e prestações de serviços das sociedades actuais, não só na área de transportes como em muitas outras.

Como se viu, os problemas no âmbito destas áreas são problemas de rentabilização dos recursos envolvidos, ou seja, de optimização ou minimização de parâmetros como o tempo, o espaço, o número de veículos, dentro dos constrangimentos próprios de cada negócio.

Como escalar, como articular, que sequência se deve seguir, para que a entrega de determinada mercadoria ou de um grupo de passageiros, ou uma sequência de tarefas industriais, que incluem produção, armazenamento e distribuição, melhor consiga atingir um objectivo ou com menos gastos.

A Logística fez surgir a Investigação Operacional e os problemas mais emblemáticos do ramo da Matemática chamado Optimização Combinatória, como sejam o PCV e o PRV. E assim, problemas práticos do dia-a-dia, de todos nós, tornaram-se problemas matemáticos, do dia-a-dia dos cientistas.

Em relação aos SIG conclui-se que, para que sejam considerados boas ferramentas para utilização na área dos transportes, deverão precisamente incluir a capacidade de modelação de redes, e funcionalidades apropriadas a operações de fluxos em redes, nomeadamente as clássicas operações de *routing* e VRP.

Em relação ao presente ‘estudo de caso’, o problema de cálculo e partição de caminhos, através de eixos lógicos e fixos de transporte, referido no estudo de caso, pode ser tratado como uma simples ‘procura do melhor caminho’ entre dois locais, dentro de uma rede de ligações existentes entre locais geográficos. Se seguidamente for extraída a sequência dos locais de passagem, sendo esses locais representados através das suas coordenadas reais, o objectivo será atingido. Intuitivamente à partida, essa extracção será facilitada pela existência de um sistema de referenciação linear.

Ou seja, apesar de este problema de transporte não depender de estradas, ele é um simples problema de análise de redes, sendo os nós dessa rede locais geográficos. Das pesquisas realizadas fica perceptível que a grande mais-valia dos Sig-T tende a estar relacionada com sistemas móveis em tempo real, e com informações muito dinâmicas que usufruirão da grande capacidade de recalcular e revisualizar rapidamente um nova solução.

Não sendo esse o caso deste ‘estudo de caso’, mesmo assim, um SIG munido de funcionalidade de *routing* e VRP, será o meio ideal para o concretizar, pois permitirá, não só, construir, como visualizar a própria rede, e posteriormente os resultados das operações sobre ela, tornando essa construção fácil, intuitiva e interessante.

Poder ainda, futuramente, aplicar a esta rede outras operações de análise de redes que possam resolver outros problemas (reutilizando) com os mesmos dados, assim como, recorrer a todas as potencialidades dos SIG clássicos, tais como, mudanças de escala, integração com outras informações, como sejam regiões administrativas, estradas, relevos, acrescentar pontos de interesse como (para além de hospitais e policias) bombas de gasolina, são argumentos de peso para a construção de uma aplicação baseada em SIG-T, para resolver o problema em questão.

### **3. PLANEAMENTO DE TRANSPORTE DE RECLUSOS**

#### **3.1 Modelação**

##### **3.1.1 Enquadramento**

Como já foi referido na introdução, uma das actividades diárias, essencial ao funcionamento da DGSP, é assegurar as deslocações necessárias, aos cidadãos em reclusão, em viatura celular, para que cumpram as obrigações de comparecer num tribunal, na polícia, uma ida a um hospital ou outro motivo.

São muitas as saídas diárias de viaturas celulares, em viagens curtas ou longas, pois muitas vezes os tribunais estão a muitos quilómetros de distância, existindo basicamente dois tipos de viagens:

1. Transporte de reclusos entre um EP e um local próximo, como seja um Tribunal, um Hospital, ou outro EP, situados num raio de não mais de 50 km, a cargo de cada EP.
2. Transporte entre dois EP, de qualquer ponto do país, a cargo de um centro de coordenação, com uma frota, para o efeito.

Em termos de serviço de transporte o ponto 1 é assegurado pelos próprios Estabelecimentos Prisionais que realizam o seu próprio planeamento e utilizam os seus próprios recursos: guardas, motoristas e viaturas.

Grande parte das viagens referidas no ponto 2, são originadas pelo facto de um recluso ter de comparecer em um tribunal que fica demasiado longe do EP em que o recluso se encontra, devendo por isso ser previamente transferido para um EP que se encontre próximo do tribunal onde o recluso deve comparecer.

Para o efeito, o TN cuja sede é em Lisboa, realiza viagens de norte a sul do país ao longo de um eixo fixo de EP.

Este serviço que será um dos focos principais deste projecto, tem a seu cargo uma média semanal de cerca de 100 transportados em viagens tipo 2, que viajam em grupo, em viaturas celulares de 16 ou mais lugares.

Esta tarefa é levada a cabo, dentro de um sistema que foi sendo concebido e melhorado empiricamente ao longo dos anos, e que conjuga a linha de transporte a cargo do TN com o transporte a cargo dos EP, da seguinte forma:

1º Foi definido um eixo principal de transporte, formado por uma linha de EP de norte a sul do país, que servem de nós de ligação entre este eixo e os eixos, que os ligam aos EP que ficam fora do eixo.

2º O eixo principal, a cargo do TN, é, por sua vez, subdividido em dois: o do Sul e o do Norte, sendo a transição de um eixo para o outro realizada em Lisboa na sede do TN.

A figura 3, mostra os eixos de transporte que foram implementados neste trabalho e reflectem o sistema de transporte.

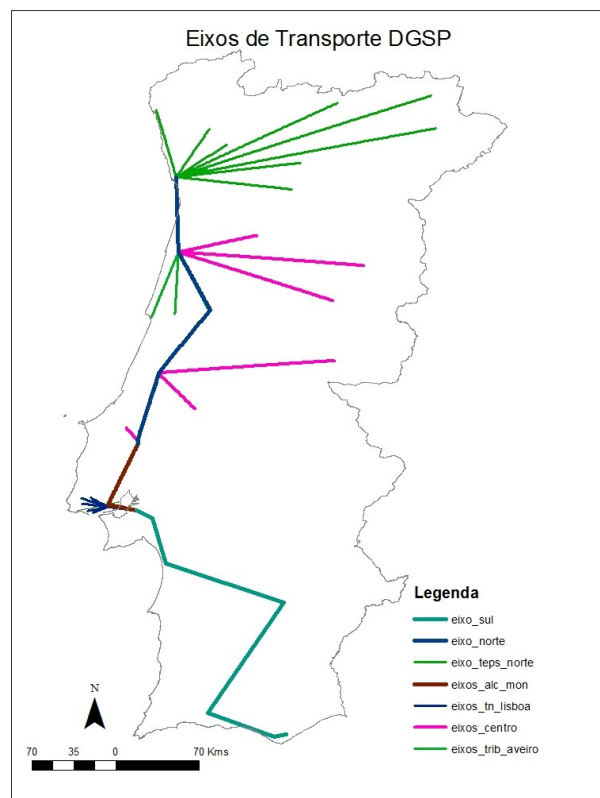


Figura 3. Mapa contendo os vários níveis de eixos de transporte da DGSP

O modo como esta rede de eixos é usada para resolver o transporte de uma

qualquer origem para um qualquer destino, pode ser ilustrado através do seguinte exemplo. Se um recluso do EPOlhão, tiver que comparecer num tribunal de Braga, ele será previamente transportado para o EPBraga, que é o EP mais próximo desse tribunal. Este processo dará origem a 4 viagens, em que a última viagem será entre o EPBraga e o tribunal, no próprio dia do julgamento:

EPOlhão -> EPLisboa (realizada pelo TN)

EPLisboa -> EPPorto (realizada pelo TN)

EPPorto -> EPBraga (realizada pelo EPBraga)

EPBraga-> tribunal de Braga (realizada pelo EPBraga)

Aqui surgem dois problemas:

- Geração de pedidos aos vários transportadores envolvidos, a partir de uma pedido inicial;
- Planeamento de viagens por parte de cada transportador.

Actualmente, o planeamento destas viagens é realizado de um modo totalmente empírico, trabalhoso e pouco eficiente, sem qualquer recurso a informática, não estando sequer os próprios pedidos de transporte informatizados, chegando sim, através de fax.

No caso dos transportadores locais o planeamento diário das viagens poderá ser muito simples ou muito complicado consoante a quantidade de pedidos diários, sendo bastante complicado, por exemplo, para os transportadores da cidade de Lisboa.

Para o TN, o planeamento das viagens longas é complicado, moroso, e totalmente manual.

### **3.1.2 Objectivos**

Na realidade, primeiro surgiu a ideia de explorar a aplicação de uma função VRP no planeamento dos vários tipos de viagens, automatizando o modo de distribuir os transportados pelos veículos, e as suas rotas (no caso do TN não há o problema de rotas, uma vez que o percurso é sempre igual).

Surgiu então o problema de não estarem informatizados os pedidos do

transportador TN, para que pudessem ser usados como input de uma ferramenta de VRP, e surgiu a ideia de automatizar o processo de geração desses pedidos, e sua consequente informatização, uma vez que, como se viu, no subcapítulo anterior as regras básicas são relativamente simples.

Uma das características principais deste problema específico, e que o diferencia da maioria dos problemas de transporte é o facto de os percursos serem fixos, e portanto, não ser necessária uma procura de caminhos físicos entre os nós, sendo que as operações de optimização seriam aplicadas a uma rede de eixos lógicos, não havendo necessidade de usar mapas de estradas o que simplificaria bastante o processo.

Os objectivos do projecto passaram a ser a criação de uma ferramenta informática com recurso a SIG, que execute geração de pedidos de transporte e geração de planeamentos de viagens, aplicados a uma rede de eixos lógicos, em que a ligação entre nós da rede significa que ‘existe transporte directo entre esses nós’, os nós da rede sendo os EP, e existindo uma ligação entre cada par de EP, sempre que exista transporte directo entre eles.

### **3.1.2.1 Geração de Pedidos de Transporte**

Pretende-se a partir da informação existente na base de dados relativa à necessidade de que um individuo compareça em determinado lugar, em determinada data, criar os necessários pedidos de transporte, e atribui-los aos respectivos transportadores, respeitando as seguintes regras básicas:

- o transporte local deverá ser assegurado pelo EP mais próximo do local de destino (excepção: zonas urbanas onde há muitos EP próximos)
- o transporte entre EP do eixo nacional é realizado através dos eixos nacionais pelo transportador Nacional. (excepção: EP da mesma região)
- se o EP de origem e/ou de destino não pertencem ao eixo nacional, o transporte entre estes e o eixo nacional é assegurado pelos EP que não pertencem ao eixo nacional.

Se considerarmos um ‘Pedido Inicial’ como um par (origem, destino), do tipo (EP, tribunal), sem qualquer relação de transporte entre eles, e um ‘Pedido Final’ um par (origem, destino), do tipo (EP, EP ou tribunal), em que a origem e o destino pertencem a um mesmo eixo, no qual o transporte é assegurado por um determinado transportador,

então pode-se afirmar que se pretende transformar um ‘Pedido Inicial’ numa sequência de ‘Pedidos Finais’.

Regras básicas de funcionamento do sistema:

- A origem do transportado é um EP, a que chamámos ‘EP origem’;
- O destino do transportado é um tribunal;
- O ‘EP escolhido’ será o EP mais próximo do tribunal;
- Pretende-se calcular o melhor caminho entre o ‘EP origem’ e o ‘EP escolhido’;
- Existem dois tipos de transportadores – TN (percorre os eixos nacionais) e EP (percorre os outros eixos);
- O transporte dentro do eixo nacional é assegurado pelo TN;
- O transporte dentro dos outros eixos é assegurado pelo EP em que se encontra o transportado;
- O eixo percorrido pelo TN está conectado com todos, os outros existindo um e um só ponto de conexão entre eles (salvo raras excepções);
- Os eixos percorridos pelos EP apenas estão conectados com o eixo do TN;
- O transporte para o tribunal é assegurado pelo ‘EP escolhido’;

A metodologia usada foi a criação de uma rede, tal com a definimos anteriormente, que integra todos os eixos de transporte da DGSP, uma vez que os pedidos podem ser entre quaisquer dois EP, e a aplicação a esta rede de uma função de ‘procura do melhor caminho’, entre a origem e o destino do ‘Pedido Inicial’.

Esta função que é normalmente aplicada a redes viárias, foi aplicada a uma rede de eixos lógicos, independentes da rede viária usada, em que o caminho resultante obtido integra uma sequência de segmentos dos vários eixos que formam a rede, pretendendo-se como objectivo final extrair esses segmentos do caminho criado.

Concretamente, a ferramenta deve procurar o nó da rede mais próximo da origem (um EP) e o nó da rede (um EP) mais próximo do local de destino e definir através dos eixos pré-estabelecidos um percurso entre os dois.

Sendo assim, definiremos uma função a que chamaremos ‘Partição Pedido’:

Descrição: Cada pedido inicial formado por um par (EP, tribunal) gera uma sequência



de pedidos finais de transporte na forma dos vários troços que são a projecção do caminho calculado nos vários eixos de transporte considerados. Usa em análise de redes um algoritmo de *routing*.

Input: par Origem-Destino;

Outputs: vários pares origem-destino; mapa com o caminho;

### **3.1.2.2 Geração de Planeamento de Viagens**

Planeamento de viagens é tipicamente um caso de PRV. Ao explorar este caso de PRV percebeu-se que não seria uma escolha de rota, ou sequência de locais, pois essa sequência é basicamente fixa, mas sim, um problema de atribuição de transportados às 4 viagens semanais com tipologias diferentes, e tendo em conta a lotação dos veículos, sabendo que nos nós da rede entram e saem transportados, e que o transporte de um transportado poderá ser numa semana ou noutra, desde que esteja no destino final na data marcada.

Para melhor compreensão do problema fez-se um estudo da aplicabilidade do PRV, não só às viagens realizadas pelo TN ao longo do eixo nacional, mas também às viagens locais, pois seria interessante que uma mesma ferramenta pudesse ser usada em diferentes tipologias de viagens.

A Viagem longa, com uma periodicidade semanal e uma duração de quase um dia, recolhe e entrega transportados. São distintos os locais de partida e chegada e é feito basicamente o mesmo percurso, em veículos de 16 ou 20 lugares. Existem 4 tipos de viagens longas: Lisboa->norte, norte->Lisboa, Lisboa->Sul, Sul->Lisboa.

Viagem curta, com uma periodicidade diária, duração de, no máximo, meio-dia, entrega e recolhe os transportados no local de destino, e regressa ao ponto de partida, depois de fazer um percurso que depende das tarefas desse dia, em veículos de 6 lugares.

Pode-se, grosso modo, falar de PRV com Entregas e Recolhas (PRVER), no caso das viagens longas, e PRV com Janelas de Tempo (PRVJT), no caso das viagens curtas.

Em ambos os casos o objectivo é calcular a melhor solução para transportar todos os indivíduos aos seus destinos (cumprir todas as tarefas da lista) usando a frota disponível. No entanto, ‘melhor solução’ no caso de viagens longas significa,

rentabilizar ao máximo a lotação do veículo, assegurando que o transportado é transportado até uma data limite e o mais próximo possível dessa data, enquanto no caso de viagens curtas ‘melhor solução’, significa ‘em menos tempo’ e ‘usando menos recursos’, mas cumprindo as restrições horárias.

A maior diferença diz respeito à tipologia da rede, à qual a função é aplicada, sendo, no caso das viagens do TN, a rede de eixos lógicos usada para gerar os pedidos, mais propriamente os eixos principais, enquanto no caso das viagens realizadas pelos EP, uma rede viária local.

Para cada viagem diária, pretende-se como resultado, para cada viagem de cada veículo uma lista ordenada de paragens, hora de chegada à paragem, informação sobre a tarefa: recolher ou entregar, e identificação do transportado.

Para cada viagem semanal, pretende-se como resultado, data da viagem, uma lista de tarefas, em que cada tarefa representa um transportado, e para cada uma delas, um local de entrada, um local de saída.

Em conclusão, no caso das viagens longas seria unicamente usada a tarefa de *scheduling* do PRV, enquanto que nas viagens locais seria usado *routing* e *scheduling*, caso típico de PRV.

Sendo assim, definiremos uma função a que chamaremos ‘Planeamento Viagem’:

Descrição: agrupa consoante as datas, a lotação da viatura e o conjunto dos pares origem-destino, os pedidos em Viagens para determinada data. Usa em análise de redes, um algoritmo de Roteamento.

Input: uma fila de pares Origem-Destino.

Outputs: uma viagem planeada, com os transportados que entram e saem em cada nó.

### **3.1.3 Construção de uma Rede de Eixos**

Uma das tarefas principais desta solução foi a construção e modelação de uma rede de eixos representativa do sistema de transporte usado.

Como foi referido atrás, cada ligação entre nós da rede significa, que ‘existe transporte entre esses nós’, mas também algumas características dessas ligações, como

por exemplo, terem apenas um sentido, vão conferir alguma complexidade necessária para modelar a realidade.

Optou-se por não criar eixos que ligam os EP aos destinos finais (tribunais), pois isso exigiria bastante trabalho, tornaria a rede demasiado cheia de eixos, e na maioria dos casos não é necessário, pois de um modo geral o transporte para o local é assegurado pelo EP mais próximo desse local, sendo a função capaz de o detectar. Nos casos de exceção terão que ser criados eixos que forcem o caminho.

Para uma maior facilidade de compreensão e posterior implementação e validação, optou-se por construir grupos de eixos com características ou funções semelhantes. Antes de mais, vão ser referidas para além do caso geral, outras situações em que o caso geral não é aplicável, e uma vez que estas exceções serão modeladas pelos próprios eixos.

### **Caso geral**

O EP mais próximo do tribunal transporta o transportado ao tribunal. Para isso o transportado, é transportado ao longo dos eixos, desde o ‘EP origem’ até esse EP.

### **Tribunais relativamente próximos do EP- exceção ao caso geral**

Quando o EP de origem não é o EP mais próximo do tribunal, mas é suficientemente próximo para que possa ser ele a transportar, deve existir uma ligação entre o tribunal e esses EP, forçando o caminho.

Exemplo1: Dentro da Grande Lisboa, cada EP leva os seus transportados aos tribunais, independentemente de serem ou não o EP mais próximo.

Exemplo2: O transporte para os tribunais de Coimbra e Figueira da Foz é realizado pelo EPAveiro;

### **Grande Lisboa: ponto de passagem**

Na sua passagem por Lisboa (onde pernoitam) os transportados que vão do norte para o sul ou vice-versa, pernoitam num dos EP, o EPLisboa, sendo este o único que liga eixos nacionais com o TN.

Através das regras de conectividade, o EPLisboa é o único que pode levar ao TN se não for origem;

### **Grande Lisboa: entrega no TN**

Outra questão refere-se à forma de entregar e de receber transportados ao TN. Aquando da partida, os EP da Grande Lisboa entregam os transportados na sede do TN. Daqui o TN dirige-se directamente ao eixo norte em EPAlcoentre ou ao eixo sul em EPMontijo. Esta situação dará origem a eixos com apenas um sentido.

No regresso o TN entrega os transportados nos EP de destino. Esta segunda situação daria origem aos eixos convenientes com apenas um sentido, mas como esses EP de destino, pertencentes à zona de Lisboa, não observam a regra do EP mais próximo, estes eixos foram postos de parte.

### **Grande Lisboa: ponto de paragem - excepção ao caso geral**

Quando um transportado de outra região do país vai a um tribunal da Grande Lisboa, não há regras fixas para a escolha do EP da zona de Lisboa, para o qual será transferido e que o levará a tribunal.

Não havendo uma regra definida, poder-se-ão construir várias soluções, mas sempre sujeitas a um ajustamento manual.

### **Grupos regionais (norte e interior) - excepção ao caso geral**

O transporte entre grupos de EP próximos uns dos outros e que não pertencem aos eixos principais, tende a ser resolvido pelos próprios. Dentro da lógica do caminho mais curto, bastará ligar esses EP entre si para que essa seja a solução escolhida. Estas ligações adequadas aos EP do centro interior e aos EP a norte do Porto não foram implementadas neste trabalho.

### **Acessos TN - excepção ao caso geral**

Alguns EP situados a norte e a sul da Grande Lisboa entregam directamente na sede do TN em Lisboa, os seus transportados que vão respectivamente para sul e norte.

## Modelação de Eixos

Foram modelados os eixos, do modo que se julgou mais conveniente seja para a sua compreensão como para a sua posterior implementação, tendo sido criados os seguintes eixos: Eixo\_Norte, Eixo\_Sul, Eixos\_Centro, Eixos\_EP\_Norte, Eixos\_Mon\_Alc\_EPLisboa, Eixos\_Lx\_TN\_Mon\_Alc, Eixos\_Tribunais\_Lisboa,

O Eixo\_Norte (figura 4) liga sequencialmente e nos dois sentidos os EP do eixo nacional entre EPAlcoentre e EPPorto, percorrido pelo TN, em duas viagens semanais, uma, no sentido EPAlcoentre - EPPorto, e outra, no sentido EPPorto -EPAlcoentre, incluindo EPAlcoentre, EPVjudeus, EPRLeiria, EPLeiria, EPCoimbra, EPAveiro e EPPorto.

As ligações entre a zona de Lisboa e EPAlcoentre, ficarão incluídas em outros eixos uma vez que são unidireccionais, e diferem consoante o sentido da viagem. Na realidade a viagem no sentido norte começa na sede do TN, localizada em Lisboa, e a viagem no sentido sul acaba nos próprios EP.

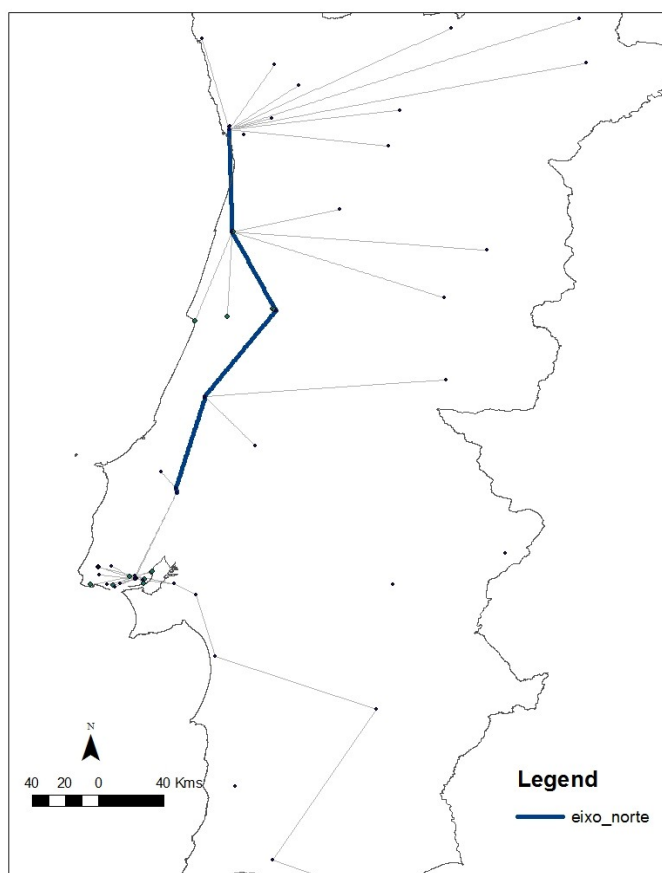


Figura 4. Eixo Norte

O Eixo\_Sul (figura 5), liga sequencialmente e nos dois sentidos os EP do eixo nacional entre EPMontijo e EPolhão, percorrido pelo TN, em duas viagens semanais, uma no sentido EPMontijo-EPolhão, e outra no sentido EPolhão-EPMontijo, incluindo EPMontijo, EPSetúbal, EPPcruz, EPBeja, EPFaro, EPSilves e EPolhão.

As ligações entre a zona de Lisboa e EPMontijo, ficarão incluídas em outros eixos, uma vez que são unidireccionais e diferem consoante o sentido da viagem. Na realidade, a viagem no sentido sul começa na sede do TN, localizada em Lisboa, e a viagem no sentido norte acaba nos próprios EP.

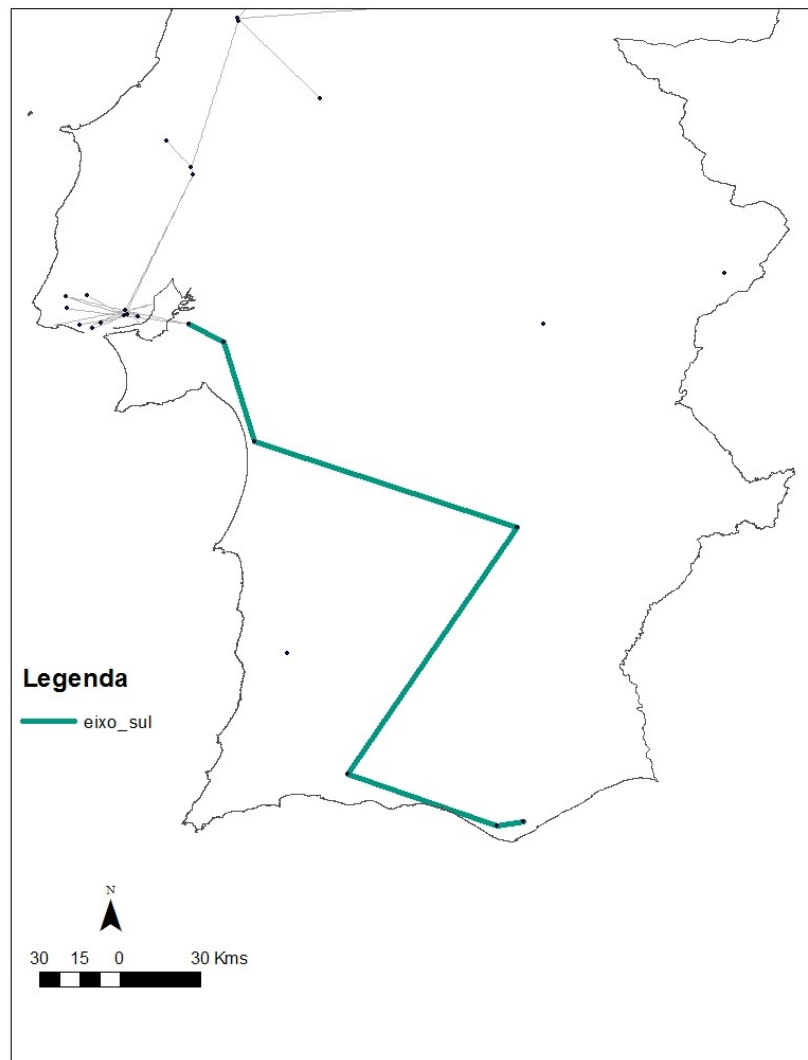


Figura 5. Eixo\_Sul

O Eixos\_Centro (figura 6), liga nos dois sentidos os EP do Eixo\_Norte aos EP situados entre Lisboa e Porto, que não pertencem a esse eixo. O transporte nestas ligações é realizado pelos próprios EP. Inclui as seguintes ligações:

EP Aveiro - EP Viseu

EP Aveiro - EP Guarda

EP Aveiro - EP Covilhã

EP R. Leiria - EP C. Branco

EP R. Leiria - EP T. Novas

EP V. Judeus - EP Caldas

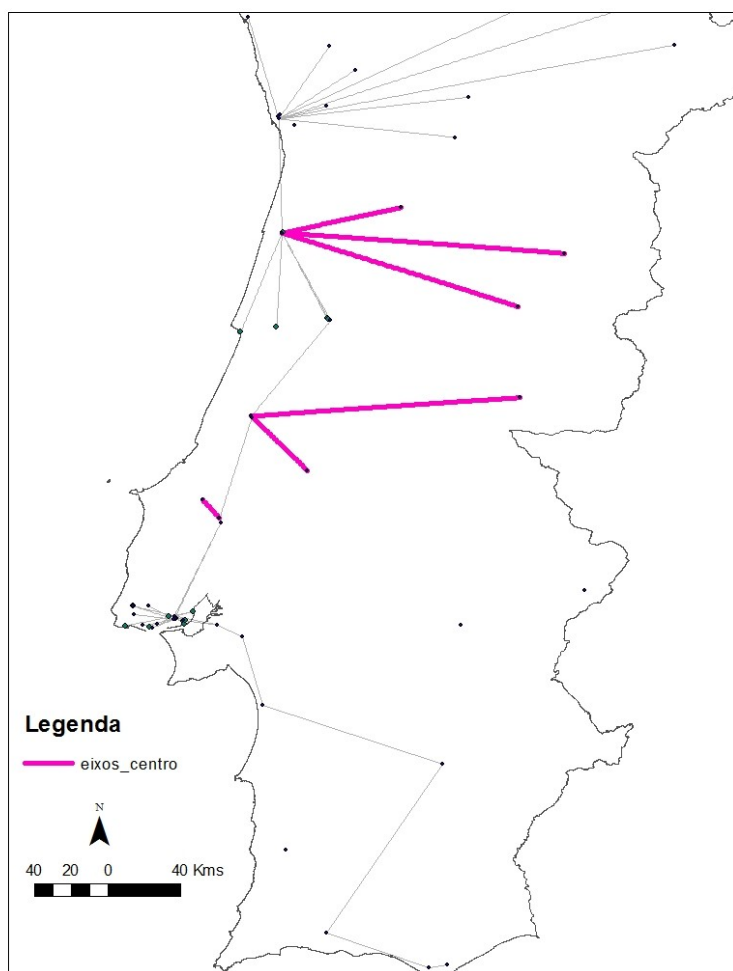


Figura 6. Eixos\_Centro

O Eixo\_EP\_Norte (figura 7) liga nos dois sentidos os EP acima do Porto, estabelecendo assim a ligação entre esses EP e o 'Eixo\_Norte'. O transporte nestas ligações é realizado pelos próprios EP. Inclui as seguintes ligações:

EPPorto - EPVcastelo

EPPorto - EPBraga

EPPorto- EPGuimarães

EPPorto - EPVreal

EPPorto - EPPferreira

EPPorto - EPSCBispo

EPPorto - EPLamego

EPPorto - EPBragança

EPPorto - EPChaves

EPPorto - EPIzeda

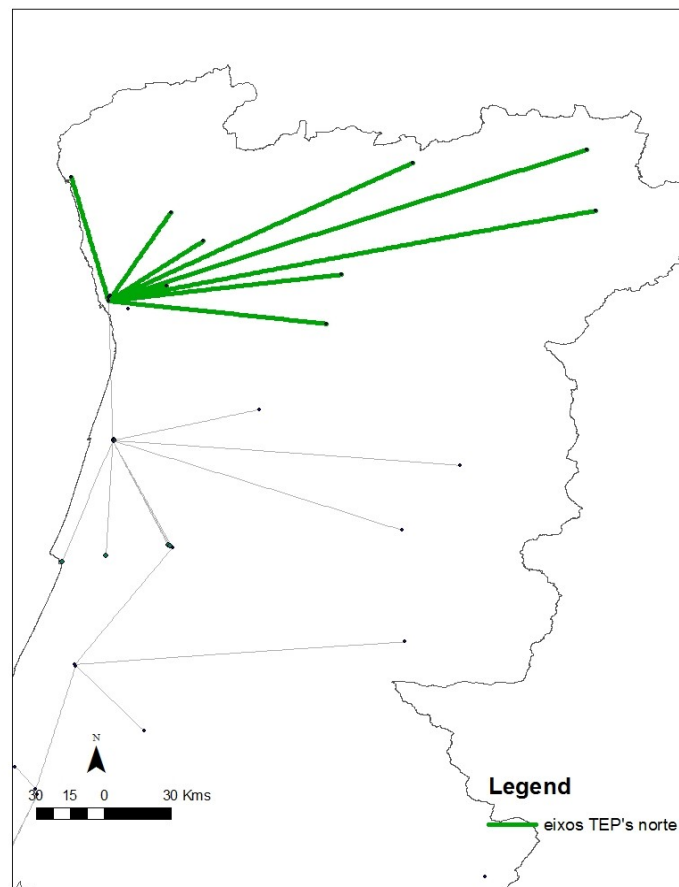


Figura 7. Eixos\_EP\_norte.



O Eixos\_Mon\_Alc\_EPLisboa (figura 8) liga em apenas um sentido EPAlcoentre e EPMontijo a EPLisboa, na continuação dos eixos norte e sul, e representando o último troço das viagens nacionais. Inclui as seguintes ligações:

EPAlcoentre -> EPLisboa

EPLisboa -> EPAlcoentre

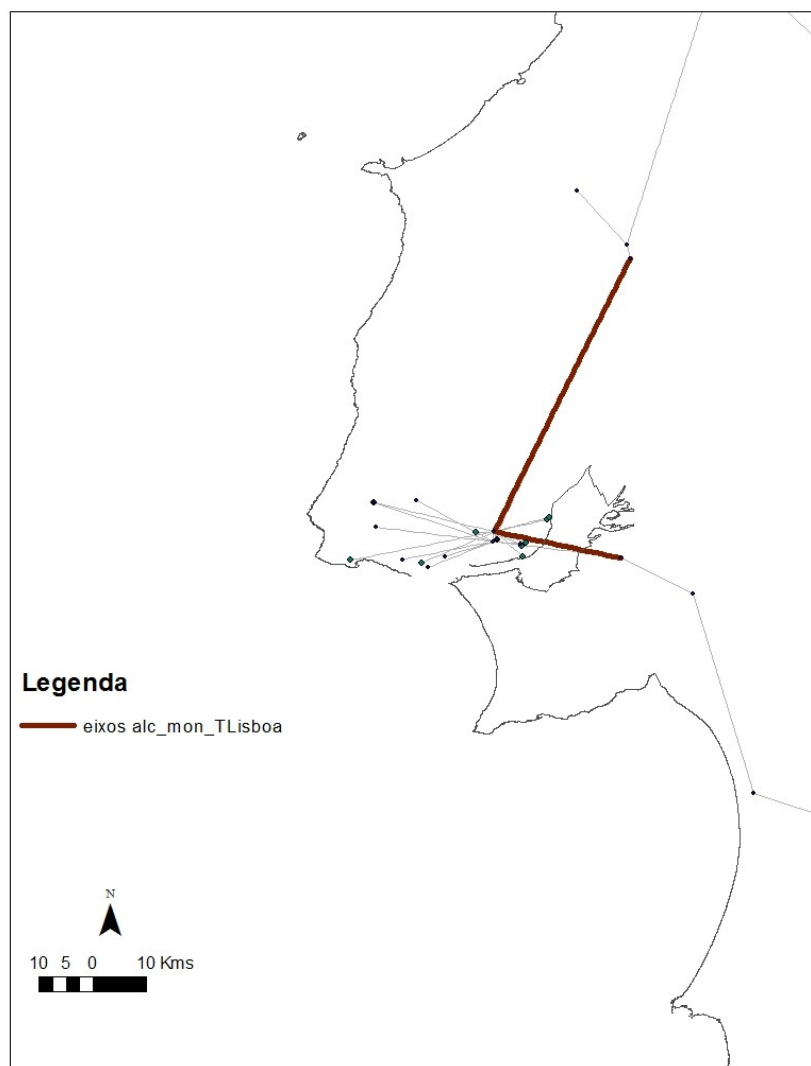


Figura 8. Eixos que ligam EPAlcoentre e EPMontijo ao EPLisboa.

O Eixos\_Lx\_TN\_Mon\_Alc (figura 9) liga, em apenas num sentido, os EP da zona de Lisboa ao TN, ligações usadas para os transportados, cuja origem são os EP da zona de Lisboa serem entregues no TN, imediatamente antes de estes serem transportados ao longo dos eixos nacionais. Inclui também ligações de um só sentido entre o TN e EPAlcoentre e EPMontijo, que são os troços iniciais dessas viagens.

TLinhó -> TN

TSintra -> TN

EPLisboa -> TN

TCargueira -> TN

TMonsanto -> TN

TCaxias -> TN

TN -> EPAlcoentre

TN -> EPMontijo

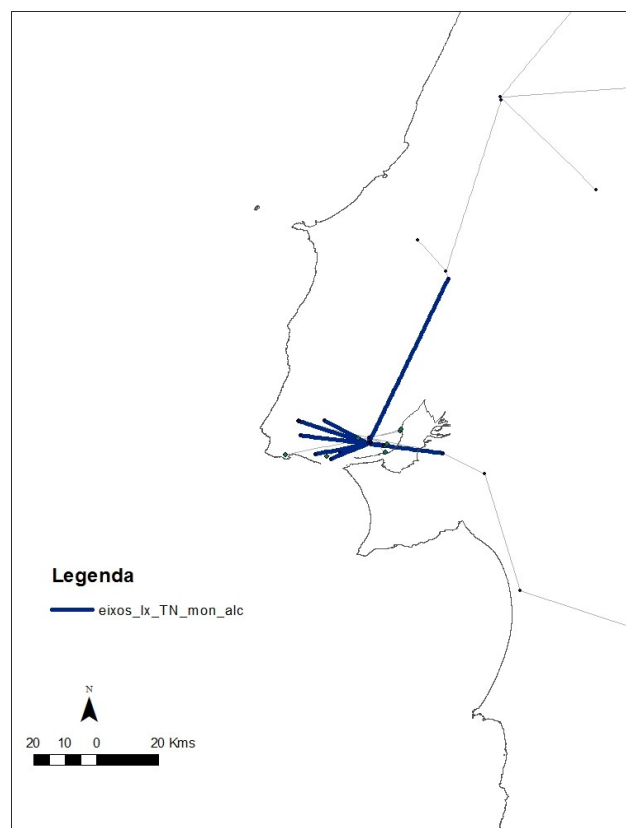


Figura 9. Eixos que ligam os EP da zona de Lisboa ao TN, e o TN a EPAlcoentre e EPMontijo.

O Eixos\_Tribunais\_Lisboa (figura 10) ligam, em apenas num sentido, os tribunais da zona de Lisboa ao EPLisboa, forçando o caminho até esses tribunais, a passar pelo EPLisboa, mesmo que na realidade não seja este o ‘EP escolhido’. Neste caso não se observa a regra do ‘EP mais próximo’ do tribunal, e não existe nenhuma regra geográfica, pré-determinada. Esta situação ficará sujeita a alterações manuais.

EPLisboa -> tribunais de Lisboa

EPLisboa -> tribunal de Oeiras

EPLisboa -> tribunal de Sintra

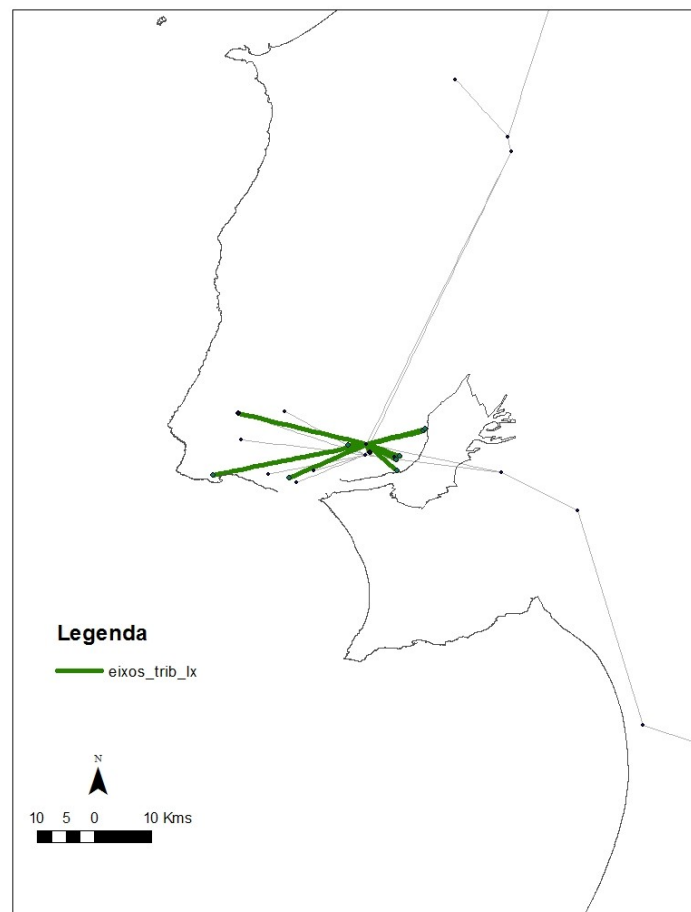


Figura 10. Eixos que ligam EPLisboa aos tribunais de Lisboa.

O Eixos\_Tribunais\_Aveiro (figura 11) liga, em apenas um sentido, os tribunais da zona de Aveiro e Coimbra, ao EPAveiro, forçando o caminho até esses tribunais, a passar pelo EPAveiro.

EPAveiro -> tribunal de Aveiro

EPAveiro -> tribunais de Coimbra

EPAveiro -> tribunal da Figueira da Foz

EPAveiro -> tribunal de Montemor-o-Novo

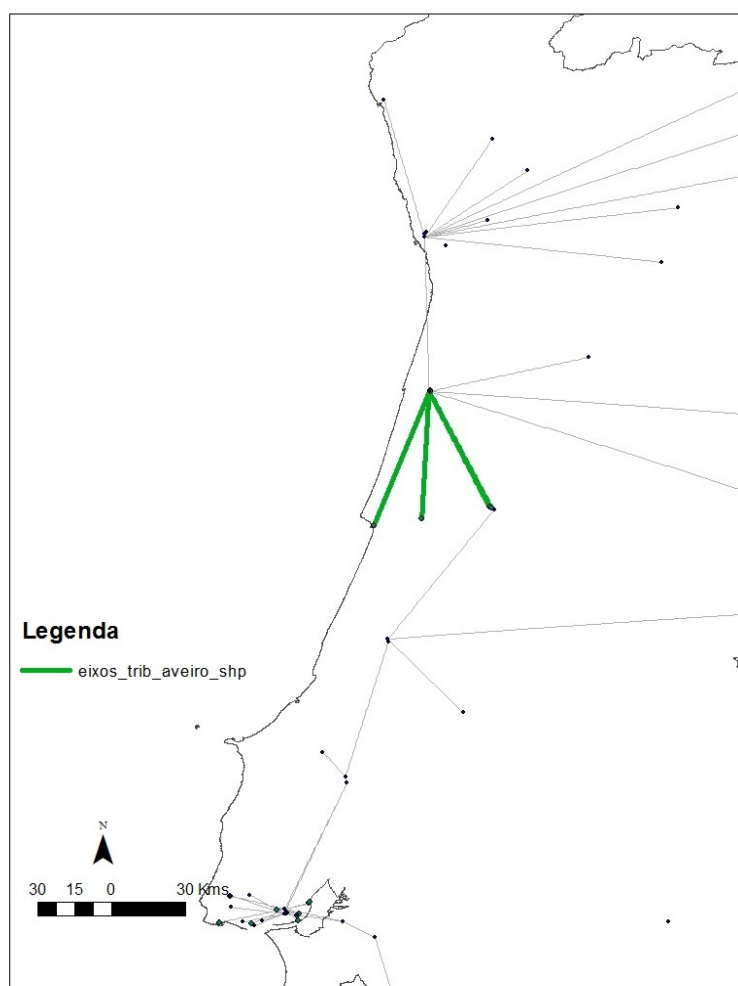


Figura 11. Eixos que ligam EPAveiro a tribunais de Aveiro e Coimbra.

O Eixos\_Liga\_TN (figura 12) liga em apenas um sentido, os EP situados entre Leiria e Pinheiro da Cruz ao TN, para serem entregues no TN, imediatamente antes de estes serem transportados ao longo dos eixos nacionais.

EPLeiria -> TN

EPRleiria -> TN

EPVjudeus -> TN

EPCaldas -> TN

EPTnovas -> TN

EPMontijo -> TN

EPSetubal -> TN

EPPcruz -> TN

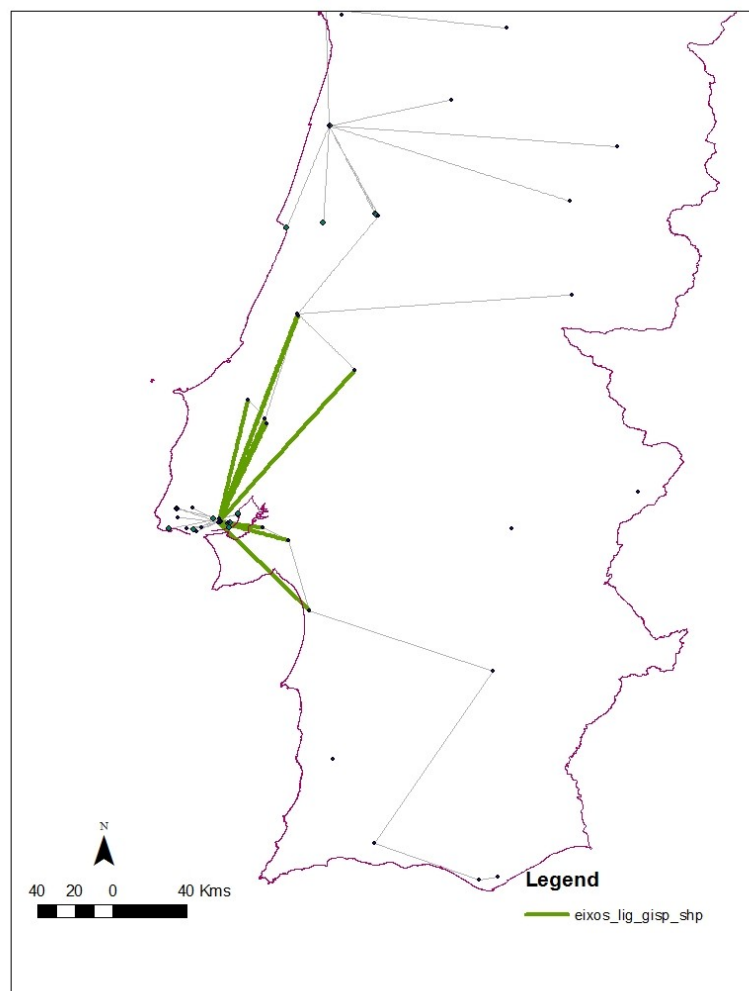


Figura 12. Eixos que ligam alguns EP ao TN.

Alguns eixos foram construídos e acabados por ser postos de parte. Foi o caso de um eixo (figura 13) para ligar EPAlcoentre e EPMontijo aos EP da zona de Lisboa, num só sentido.

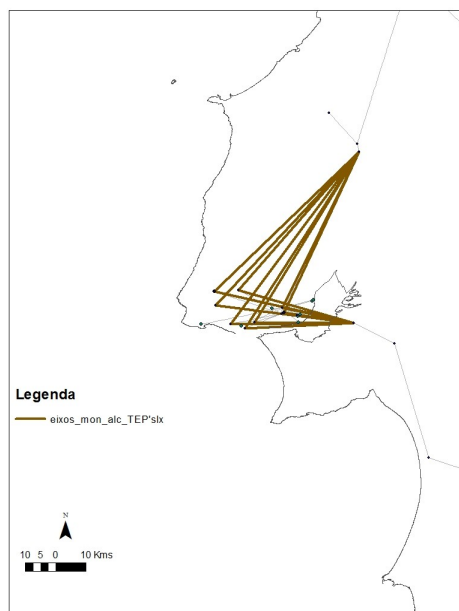


Figura 13. Eixos que ligam os EP da zona de Lisboa a EPAlcoentre e EPMontijo.

Foi também o caso de um eixo (figura 14) para ligar os tribunais de Lisboa aos EP de Lisboa, num só sentido.

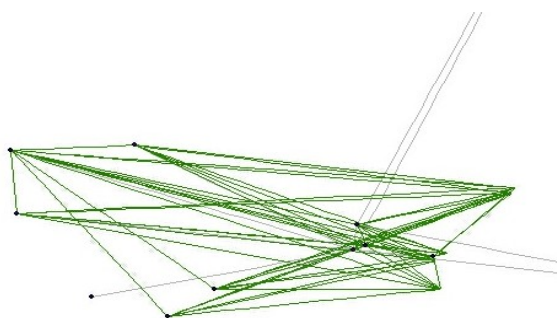


Figura 14. Eixos que ligam os EP de Lisboa aos tribunais de Lisboa.

## Redes de eixos

Dentro da terminologia de redes, os nós das redes são os EP, a sede do TN e alguns tribunais, e as arestas, as ligações entre eles, já modeladas através dos eixos previamente criados.

Apesar de, como resultado final, terem sido construídas duas redes, a ‘Rede2’ e a ‘Rede3’, considera-se que foram construídas três redes, uma vez que a primeira, a que se designou de ‘Rede1’, e que foi transformada na ‘Rede2’, foi a base de que se partiu para construir as outras.

**‘Rede1’** - criada com o objectivo de resolver as regras básicas e mais usadas..

- **Eixos** - Formada pelos eixos nacionais e ligações entre estes e os outros transportadores. Todos os EP da zona de Lisboa, são ligados ao EPLisboa.
- **Nós** – os EP.
- **Conectividade** - um único grupo formado por todos os eixos e os EP.
- **Atributos** - distância, tempo, nome.
- **Vantagens** – simplicidade, pois resolve tudo através do caso geral.
- **Desvantagem** - não resolve as situações em que não é usada a regra do ‘EP mais próximo’. Fica de fora a entrega no TN na zona de Lisboa.

**‘Rede2’** – Criada com o objectivo de refinar a rede1, acrescentando-lhe a capacidades.

- **Eixos**: aos da ‘Rede1’, foram acrescentados os eixos tribunais da zona de Lisboa e de Aveiro e as ligações ao TN na zona de Lisboa., substituíram a anterior ligação ao EPLisboa. Foram criados, mas depois postos de parte eixos entre o EPAlcoentre e o EPMontijo e os EP da zona de Lisboa, e entre estes e todos os tribunais da zona de Lisboa.
- Desistiu-se do eixo que liga o EPAlcoentre e o EPMontijo os EP de Lisboa.
- **Nós** - os EP, a sede do TN, os tribunais da zona de Lisboa, de Aveiro e Coimbra.
- **Conectividade**: um único grupo formado por todos os eixos, e todos os nós.
- **Atributos**: distância, tempo, nome, *oneway* (restringindo um dos sentidos dos troços que formam os eixos).
- **Vantagens**: Melhora a zona de Lisboa e as transferências para o EPAveiro.
- **Desvantagens**: As viagens locais de Coimbra passarão a ir a Aveiro.

**‘Rede3’** – Inicialmente criada com o objectivo de refinar a ‘Rede2’, acrescentando-lhe mais uma potencialidade, acabou por ser uma rede que deverá ser usada separadamente, em casos específicos.

- **Eixos** – ‘Rede2’, acrescentada do Eixo\_lig\_TN, que liga os EP entre EPRleiria e EPPcruz, inclusive, directamente ao TN.
- **Nós** - os mesmos da ‘Rede2’.
- **Conectividade** - criação de dois grupos de conectividade. Os novos eixos com ligação directa ao TN, não estão conectados com os eixos nacionais. Os EP foram separados em dois grupos: os que ligam directamente ao TN (menos os de Lisboa) e os outros. O TN liga os dois grupos de conectividade.
- **Vantagens** - Resolve correctamente o que se pretendeu resolver: que apenas os pedidos com origem nos EP entre EPRleiria e EPPcruz, tenham acesso ao novo eixo criado, que é o caminho mais curto para o TN.
- **Desvantagens** - Os caminhos dos pedidos cujo ‘EP origem’ está fora da área entre EPRleiria e EPPCruz, e o ‘EP escolhido’ se encontra dentro dessa zona, não conseguem ser calculados.

### 3.1.4 Modelação Geral do Sistema

As duas funcionalidades principais Partição de Pedidos e Planeamento de Viagens, foram implementadas em Python depois de devidamente exploradas no ArcGIS. O output da primeira é o input da segunda (figura 15).

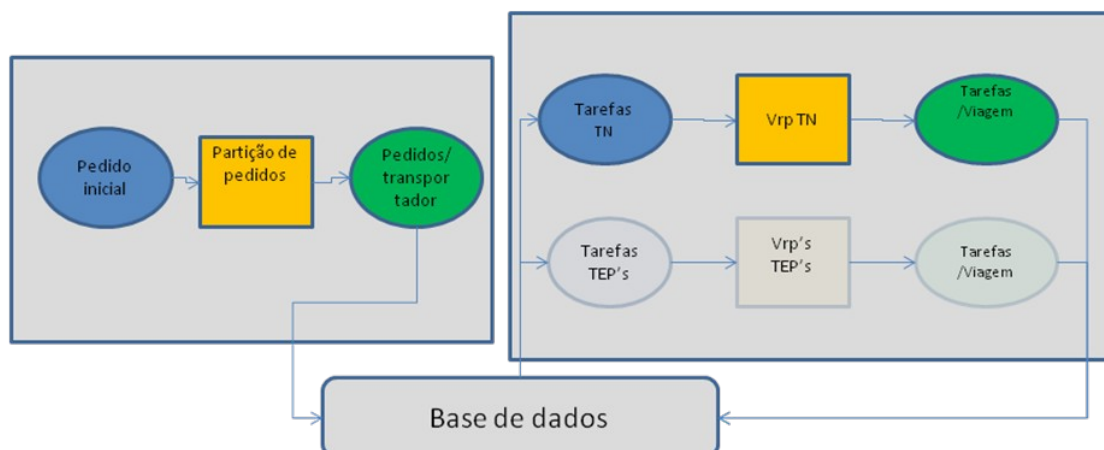


Figura 15. Esquema geral das funcionalidades principais



### 3.1.5 Obtenção de Dados

Todos os modelos de dados espaciais, usados neste projecto, são vectoriais como é normal em análise de redes

Para criação da rede foram criados de raiz as *shapes* necessárias. Na criação das *shapes*, foi usada a projecção Geográfica, WGS84, compatível com a do Google Earth. As coordenadas dos EP foram extraídas através do Google Earth e do software Geocoder1.0. As coordenadas dos tribunais foram obtidas através do software Geocoder1.0.

Foi também usado um mapa com os limites de Portugal para enquadramento geográfico da rede criada. Foi usada a Carta Administrativa Oficial de Portugal (CAOP) versão 2009, retirada do *site* do Instituto Geográfico Português.

## 3.2 Desenvolvimento

### 3.2.1 Introdução

O software desenvolvido consiste essencialmente em duas partes, a do ArcGIS/Python e a do Oracle. A primeira executa funções SIG, as funções principais do projecto, e a segunda fornece os dados e recebe e valida os resultados. A passagem de um contexto para o outro não é automática sendo realizada manualmente através da passagem manual de dados de Oracle para Excel e vice-versa.

### 3.2.2 Softwares

Foram usados os *softwares*, ArcGIS 9.3.2 e ArcGIS 10, Python 2.5 e Python 2.6, Excel97 e Excel2007, o GoogleEarth e o GeocoderV1.0. um, para realizar as funções geográficas, um de *script*, que permite facilmente juntar e correr um conjunto de operadores geográficos, um gestor de base de dados Oracle, e um software auxiliar para passagem de um para o outro, uma vez que não foi realizada conexão directa entre eles. A figura 16, ilustra a ligação entre os softwares.usados.

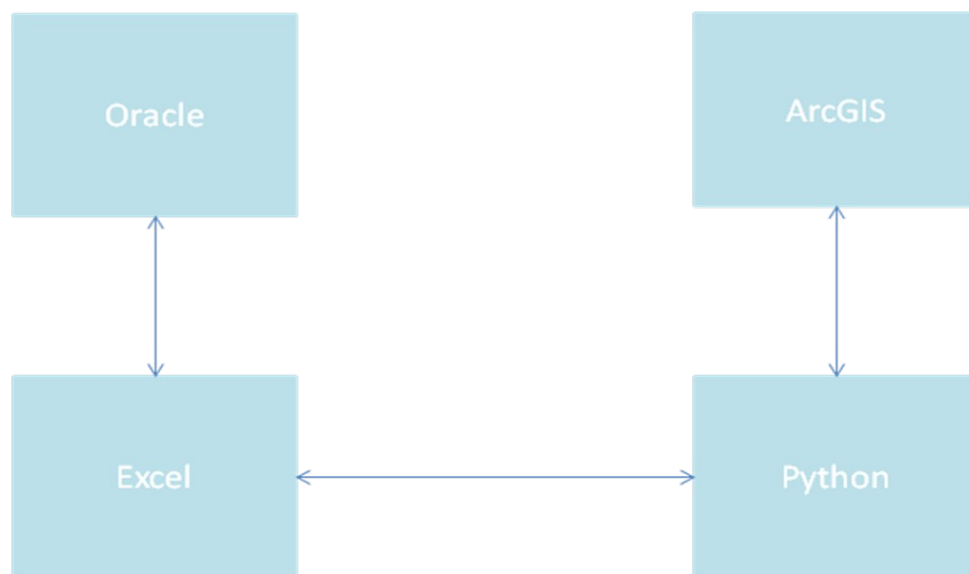


Figura 16. Esquema de interligação dos vários *softwares* usados

Para realizar as funções geográficas foi usado o ArcGIS 9.3.2., principalmente a extensão, Network Analyst.

O ArcMap foi usado para construir as *shapes* dos eixos, e realizar uma primeira exploração de todas as funcionalidades que depois foram chamadas através do ficheiro Python.

O ArcCatalog foi usado para construir as redes, dentro de *geodatabases* contendo as *shapes* que formam os eixos.

Os scripts com as funcionalidades principais, foram criados em linguagem Python, nas versões Python 2.5 e Python 2.6, editada através do IDLE - *Integrated Development Environment*, que é o *software standard* de desenvolvimento em Python.

Foi usado o gestor de base de dados Oracle 10g, através da linguagem SQL – *Structured Query Language*, para criar as tabelas e procedimentos de acesso à base de dados.

Para passar dados, da base de dados Oracle para o ArcMap e Python foi usado o Microsoft Excel.

O Google Earth foi usado para obter as coordenadas geográficas dos EP.

Para obter a coordenadas geográficas dos tribunais foi usado o GeocoderV1.0 que é um *software* disponível na Internet, e que permite obter as coordenadas através de códigos postais e moradas.

## Linguagem de Programação Python

A linguagem de programação de alto nível Python foi lançada em 1991 por Guido van Rossum, na Holanda, tendo o nome sido inspirado no nome do grupo Monty Python, de que Guido van Rossum era fã.

Segundo palavras do seu criador, Python combina elementos dos maiores paradigmas de programação (procedural, funcional e orientada a objectos), com uma sintaxe elegante, fácil para os olhos e fácil de usar.

Na realidade todos os que se referem a Python, realçam a sua simplicidade e consequente facilidade, assim como a sua flexibilidade e capacidade de integrar (e integrar-se) em vários ambientes e linguagens, e ainda por cima, podendo ser carregada livremente a partir da Web.

O padrão *de facto* é a implementação Cpython. Cpython é a implementação principal da linguagem de programação Python, escrita em linguagem C.

Existem vários ambientes de desenvolvimento integrado (IDE) disponíveis para Python, como por exemplo o IDLE, usado neste trabalho desenvolvido pelo próprio Guido van Rossum e sendo actualizado para cada nova versão de Python.

É aplicada em domínios científicos como bioinformática, física, no desenvolvimento de jogos, e em muitas soluções para Internet.

No que diz respeito à integração Python/ArcGIS, o principal objectivo do geoprocessamento é automatizar uma tarefa que conjuga várias funcionalidades SIG. O Python é uma das linguagens de *script*, através das quais se pode aceder às funcionalidades do ArcGIS, para com elas criar novas ferramentas. Linguagens de *script* são baseadas em texto e editadas em texto.

A ligação entre os dois ambientes ArcGIS e Python é realizada através da criação de um objecto o ‘geoprocessor’ que serve de veículo entre o ArcGIS e a linguagem de *script*, sendo as funções de geoprocessamento do ArcGIS usadas como métodos do referido objecto.

O Python2.5 vem com o ArcGIS 9 e é instalada juntamente com as componentes típicas de uma instalação. Com o ArcGIS 10, é instalada a Python versão 2.6., e o ‘site-package’ *ArcPy*. ‘Site-package’ é o termo usado em Python para referir uma biblioteca de rotinas que estende a linguagem Python com funcionalidade adicional, ou seja uma

extensão e é o sucessor do módulo *arcgisscripting*.

As características gerais são a facilidade, simplicidade e flexibilidade. Segundo (Honeycutt, 2010), a linguagem Python é:

- Fácil de aprender;
- Fácil de usar como todas as linguagens de *script*;
- Excelente para principiantes e especialistas;
- Adequado a grandes ou pequenos *scripts*;
- É interpretada, não necessitando portanto de compilação;
- É *Open Source*.

Não deixando no entanto, de:

- Ser uma linguagem de alto nível;
- Com todas as capacidades de programação de uma linguagem de desenvolvimento completa;
- Muitos tipos de dados (simples e estruturados como sejam listas, dicionários, etc.);
- Suporta programação orientada a objectos.

E ainda as suas capacidades de integração:

- Multi-plataforma: Windows e Linux;
- A sua fácil integração com outras linguagens revela que é um sucesso a agregar diferentes sistemas: Integrável com Java, Fortran e C++;
- ODBC interfaces com vários S.G.B.D. como Oracle, e Sql Server;
- Interacção com Web Services e bases de dados.

### **3.2.3 Desenvolvimento**

O desenvolvimento incluiu 4 tarefas principais:

- Criação de uma rede em ArcGIS.
- Construção da ferramenta Partição de Pedidos em ArcGIS/Python

- Construção da ferramenta Planeamento de Viagens em ArcGIS/Python
- Procedimentos Oracle

### 3.2.3.1 Criação de uma Rede

**Descrição:** Foram criadas *Shapes* de linhas contendo os eixos que irão integrar a rede, tendo estas por sua vez sido construídas a partir das coordenadas de pontos correspondentes aos locais dos EP e tribunais. Na construção da rede deve também ser considerada a ‘conectividade’ e a ‘impedância’. O tipo de ‘conectividade’ define os nós de ligação entre eixos, e a ‘impedância’ define o valor que será usado na avaliação do melhor caminho.

**Input:** ficheiros de texto contendo as coordenadas.

**Output:** um objecto do tipo ‘rede’.

**Métodos:**

*Create Feature From Text File* – cria uma *feature class* baseada em coordenadas contidas no ficheiro de texto; existe um parâmetro que indica qual o separador decimal, neste caso uma vírgula

*XY to Line* – Usada para substituir a anterior que foi descontinuada no ArcGIS10 – o input deste método é um ficheiro Excel.

*CreateNetwork* – São inseridas na rede as *shapes* de linhas contendo os eixos e a *shape* pontos contendo os EP, definida a: conectividade entre os eixos, e os parâmetros, impedância (foi usada a distância entre nós), e sentido únicos nalguns eixos.

*Build Network* do *Network Analyst* – depois de definida a rede tem que ser construída, de acordo com os parâmetros e variáveis definidos, sendo criados dois novos objectos a ‘rede’ e a uma *feature class* de pontos contendo as junções entre eixos caso não existam na *shape* de pontos.

A figura 17 mostra o modelo que foi usado para construção das redes. O modelo apenas inclui alguns eixos, os primeiros a serem construídos numa rede experimental, e não refere alguns parâmetros como sentido único, mas consegue representar o modelo de construção.

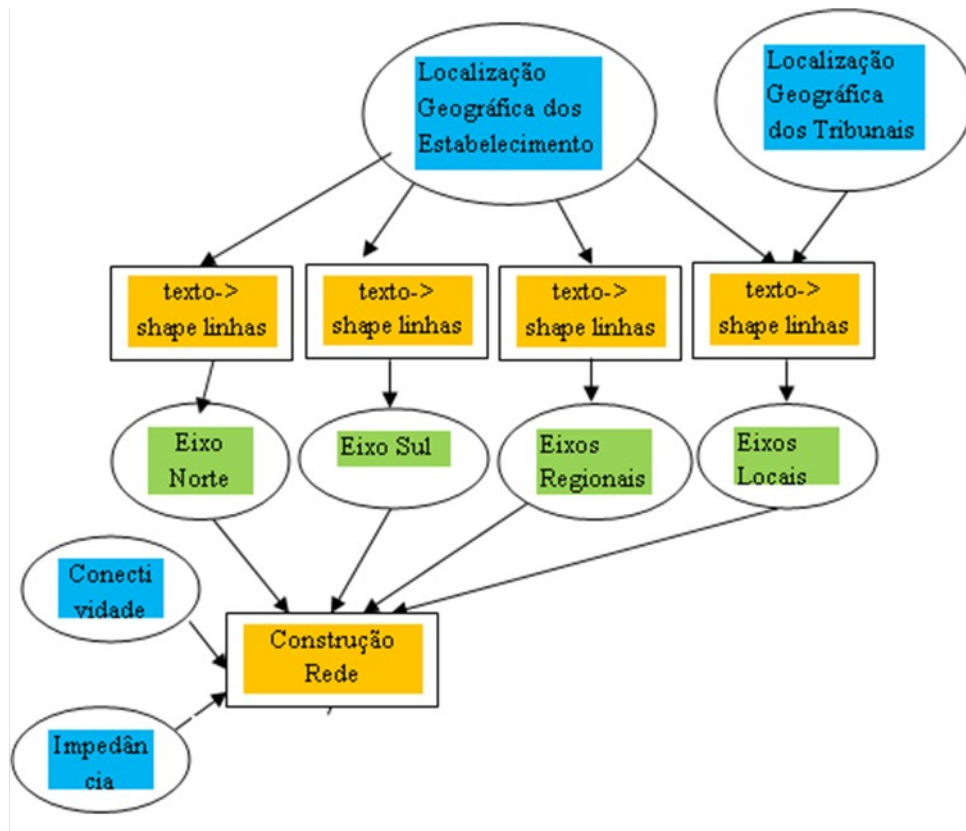


Figura 17. Modelo de construção de uma rede de eixos.

### 3.2.3.2 Partição de Pedidos

Esta ferramenta fundamental neste projecto, implementada em ArcGIS/Python usa uma sucessão de operadores do ArcGIS alguns do *Network Analyst*, outros não, para proceder à transformação de um ‘pedido inicial’ em vários ‘pedidos finais’. A ideia para criar esta ferramenta surgiu devido ao facto de esses pedidos ‘finais’ não estarem informatizados, sendo de uma enorme mais-valia a sua geração automática a partir de pedidos iniciais.

Baseia-se na função *Route* aplicada a uma rede de eixos predefinidos, e que encontra o melhor caminho, neste caso o único, entre dois pontos. Destes dois pontos, o de partida pertence obrigatoriamente ao eixo, e o de chegada pode não pertencer ao eixo procurando o ponto do eixo mais próximo. Depois de gerado o percurso, ele é intersectado com os nós que definem os eixos, gerando a lista dos pontos de passagem.

A lista resultante será posteriormente, processada em Oracle para dela serem extraídos apenas os extremos de cada eixo, tal como se pretende. Ir-se-á descrever o processo que recebe um ficheiro Excel contendo para cada pedido as coordenadas de um EP e as coordenadas de um tribunal, e devolve um ficheiro de extensão \*.dbf, contendo para cada um desses pedidos, uma sequência de EP, desde o ‘EP origem’ até ao ‘EP escolhido’.

**Input:** um Excel contendo uma lista de pedidos duplos: a cada pedido correspondem dois registos, com o mesmo identificador, cada um, com um par de coordenadas.

**Output:** um ficheiro de extensão .dbf contendo tantas linhas por pedido quantos os locais de passagem.

#### **Métodos:**

Aqui os métodos são descritos de modo sequencial, tal como são chamados no ficheiro Python e ilustrados na figura 18:

*Make Route Layer* – cria um ambiente de análise para *routing*, associado à rede de eixos.

*MakeXYEventLayer* - Cria um *layer* de pontos, baseado nas coordenadas x,y contidas no

ficheiro Excel. Passa também um parâmetro contendo informação relativa à projecção geográfica.

*AddLocations* – Fornece os ‘locais’ para a análise de *routing*, neste caso pares ordenados de pontos. Os parâmetros mais significativos são *search\_tolerance* que foi colocada a 50000 metros para garantir a distância entre um tribunal e um EP e o *match\_to\_closest* que define que se pretende o EP mais próximo.

*Solve* – executa a análise apropriada a um Route layer, ou seja aplica a função de *routing* aos pares origem-destino, gerando uma *route* (um caminho através dos eixos) para cada par.

*Copy Features* – Cria uma *shape* de linhas, contendo uma linha para cada *route* resultante da análise (*solve*), para que possam ser usadas na funcionalidade seguinte.

*Locate Features Along Routes* – calcula a intersecção entre cada *route* e a *shape* de pontos contendo os EP, gerando uma tabela, contendo, para cada *route*, uma lista de EP, caracterizados como eventos na *route*, ou seja associados a uma coordenada linear, que é a sua distância ao início da *route*. Através desta coordenada é possível ordenar essa lista como uma sequência, a sequência de EP, desde o EP origem’ ao ‘EP escolhido’. Os parâmetros mais significativos são *route\_id\_field*, que define qual o campo que identifica unicamente cada *route*, neste caso ‘Name’, *radius\_tolerance* que indica a distância tolerável da *route* aos eventos que se querem detectar, neste caso os EP que pertencem à *route*, e portanto o parâmetro assume o valor ‘ZERO’, *route\_locations* que determina se apenas se pretende o primeiro evento encontrado ou todos, usando o valor ‘ALL’, *distance\_field* determina se se pretende a distância do evento à *route* (em caso de ‘NO\_ZERO’) usado com o valor ‘NO\_DISTANCE’, *in\_fields* indica se pretende que a tabela resultante inclua os campos associados aos EP presentes na tabela de input usado com ‘NO\_FIELDS’, (ou ‘FIELDS para teste e apresentação dos resultados)

A figura 18, ilustra o fluxo de operações, que foi implementado, em Python. Nesta imagem criada no *model builder*, não está representada a ligação entre *Route 2(3)* no



fim primeiro bloco e *Route/Routes* no início do segundo, sendo *Route/Routes* um sub-layer de *Route 2(3)*.

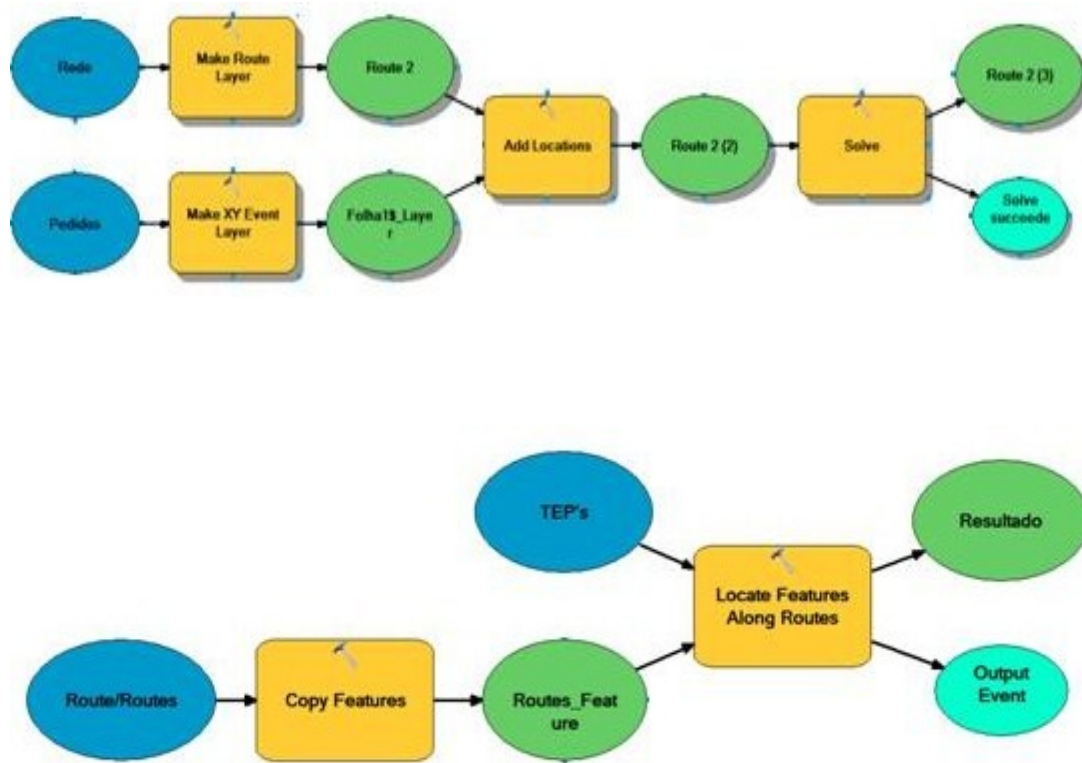


Figura 18 – Fluxo de Operações da funcionalidade ‘Partição Pedidos’.

### 3.2.3.3 Procedimentos Oracle

Uma fase importante no desenvolvimento do projecto foi o trabalho realizado ao nível da base de dados Oracle (SIP), necessário para preparar os dados para serem usados pelas ferramentas assim como receber os resultados e integrá-los no contexto hoje existente.

Para Input, Output e Validação de Resultados, foram criadas as seguintes tabelas ou views: a tabela *TFROPINI*, que armazena informação referente aos pedidos iniciais, a tabela *TFROCOLO*, que guarda as coordenadas de EP e tribunais, a view *VFROPEDU*, que usa as tabelas anteriores para criar para cada pedido dois registos com as coordenadas da origem e do destino, a tabela *TFROPEPA*, que armazena as sequências de EP resultantes do script Python e a tabela *TFROPEFI*, que armazena os pedidos finais um por transportador. Foram também criados os seguintes métodos:

*Pedidos\_Iniciais* – passa os pedidos de uma tabela já existente, na base de dados, para a tabela criada para o efeito *TFROPINI*, acrescentando-lhe um campo (PINI\_ID) que os vai identificar unicamente durante todo o processo. A partir dos dados inseridos na tabela *TFROPINI*, foi criada a *view VFROPEDU* que transforma cada registo correspondente a um pedido inicial, em dois registos, um com as coordenadas da origem, e outro com as coordenadas do destino que partilham o mesmo identificador, e que servirá de input ao ficheiro Python.

*Tratar\_Pedidos\_Partidos* - este procedimento vem completar o tratamento já dado ao pedido inicial. A partir dos dados passados do ficheiro .dbf resultante da funcionalidade *PartiçãoPedidos*, para a tabela *TFROPEPA*, este procedimento retira os EP intermédios de cada eixo, para cada grupo de pedidos, e ainda os transforma em pedidos com origem e destino e insere os novos pedidos (partidos) na tabela *TFROPEFI*. Nesta tabela já se consegue visualizar o resultado da aplicação da *Partição*.

*Inserir\_Pedidos\_Partidos* - procedimento que passa os dados para as 3 tabelas já hoje existentes na base de dados do SIP\_Frotas que contêm informação relativa a Pedidos de transporte, Origens-Destino e Transportados.

*Valida\_Resultados*- procedimento criado para apurar a aplicabilidade da ferramenta confrontando dados obtidos com dados reais.

### **3.2.3.4 Procedimentos Manuais**

*Do\_Oracle\_Pro\_Excel* - o resultado da *view VFROPEDU* é gravado manualmente num ficheiro Excel que servirá de input à funcionalidade *PartiçãoPedidos* executada através do ficheiro Python.

*Do\_Dbf\_Pro\_Oracle* – o ficheiro *dbf* resultante da funcionalidade *PartiçãoPedidos* é aberto no *Microsoft Excel* e a partir dessa informação são manualmente criadas linhas de *SQL* que vão ser usadas para inserir a informação na tabela *TFROPEPA*.

O esquema seguinte (figura 19) engloba a *Partição de Pedidos* no seu todo, incluindo os procedimentos manuais (a tracejado) e os procedimentos em Oracle:

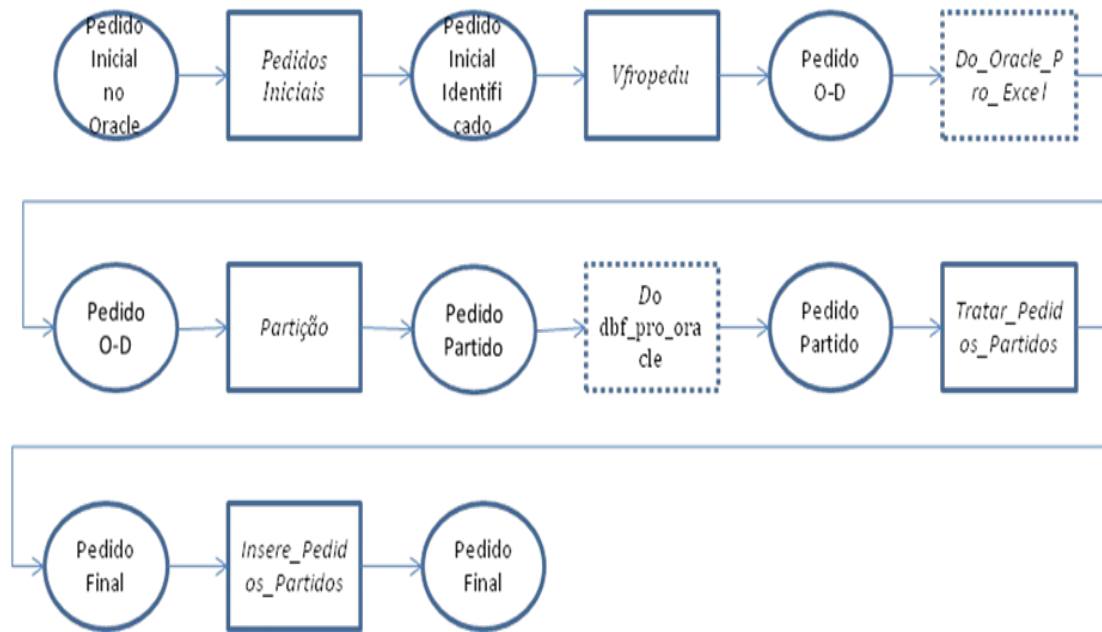


Figura 19. Fluxo Geral de 'Partição Pedidos'

### 3.2.3.5. Planeamento de Viagens

Para planeamento de viagens foi usado o layer de análise Vehicle Routing Problem, do Network Analyst. Depois de explorado todo o processo no ArcGIS, foi criado o correspondente script Python. Apesar de o objectivo ser o planeamento das viagens longas, para uma melhor compreensão das possibilidades da análise, foi realizado um estudo comparativo da parametrização, para viagens longas e para viagens curtas, que está resumido na tabela 4. Uma vez que a construção desta análise é algo complexa, a tabela 3 resume os parâmetros mais importantes no contexto deste trabalho, em cada um dos sub-layers, do VRP layer.

A tabela 2 refere os materiais usados:

Nome/Descrição	Atributos Espaciais	Atributos não Espaciais	Tipo de ficheiro
Pedidos	Localização da Origem, Localização do Destino	Data/Hora no local	Ficheiro de texto
Frota	Local de garagem	Matricula/Viagem, Lotação	Ficheiro de texto
Rede de Eixos Lógicos de Transporte	Localização de Edges e nós da rede	Tempo, Distância	Ficheiro de extensão .ND

Tabela 2. Materiais usados em VRP

Um *layer* para uma análise de VRP inclui várias classes de features, das quais indico as usadas neste modelo, e para cada uma delas, na tabela 3, são referidos os atributos significativos para este modelo. Estas *shapes* e tabelas foram criadas a partir dos ficheiros de texto referidos na tabela anterior.

**Order:** *shape* de pontos que representa neste contexto a necessidade de levar/trazer uma pessoa a/de um local. A geometria do ponto representa a localização desse local.

**Route:** *shape* de linhas, inicialmente vazia apenas com alguns atributos iniciais, sendo a sua geometria criada durante o processo; cada feature representa a viagem de uma viatura. A geometria da *route* é o caminho percorrido.

**Depots:** *shape* de pontos que representa as garagens das viaturas; as viagens começam e acabam numa destas garagens. A geometria do ponto é a localização da garagem.

**Order Pairs:** tabela que estabelece uma relação sequencial entre duas ‘orders’; é usada para os casos, como seja, este em que o transportado vai de um local para outro, não sendo ‘carregado’ obrigatoriamente no destino; também serve para criar regressos de um local. Nestes casos são necessários dois pares. A cada ‘levantamento’, corresponde uma ‘entrega’.

<b>Order</b>	<b>Inputs não espaciais:</b> Identificador ( <i>objectoid</i> ), Nome do destino da tarefa ( <i>name</i> ), Tempo no serviço ( <i>Service Time</i> ), Hora mínima de atendimento ( <i>TimeWindowStart</i> ), Hora máxima de atendimento ( <i>TimeWindowEnd</i> ), Tempo gasto no local ( <i>ServiceTime</i> ) Nº de pessoas que Entram ( <i>PickupQuantity</i> ), Nº de Pessoas que Descem ( <i>DeliveryQuantity</i> ), Violação Máxima de Tempo ( <i>MaxViolationTime</i> ), Eixo em que se situa ( <i>SourceId</i> ).	<b>Outputs não espaciais:</b> <i>Route</i> /Viatura na qual foi executada/transportada ( <i>RouteName</i> ), Ordem de execução dentro da sequência ( <i>Sequence</i> ), Data/hora de chegada ao local ( <i>ArriveTime</i> ), Data/hora de partida do local ( <i>DeparTime</i> ).
<b>Route</b>	<b>Inputs não espaciais:</b> Identificador ( <i>objectoid</i> ), Nome da viatura/viagem ( <i>name</i> ), Depósito de partida ( <i>startdepot</i> ), Depósito de chegada ( <i>Arrivaldepot</i> ), Data/hora mínima de partida ( <i>EarliestStarTime</i> ), Data/hora Máxima de partida ( <i>LateStarTime</i> ), Lotação de veículos ( <i>Capacities</i> ), Tempo Máximo ( <i>MaxTotalTime</i> )	<b>Outputs não espaciais:</b> (apenas existe output quando a <i>route</i> /viatura foi utilizada) nº de tarefas executadas ( <i>OrderCcount</i> ), tempo total ( <i>TotalTime</i> ), hora de partida ( <i>start time</i> ), hora de chegada ( <i>end time</i> ), tempo total ( <i>TotalTime</i> ), tempo em andamento ( <i>TotalTravelTime</i> )
<b>Depot</b>	<b>Inputs não espaciais:</b> Identificador ( <i>objectoid</i> ), Nome ( <i>name</i> ), Eixo da rede em que se localiza ( <i>SourceID</i> ) Hora de abertura ( <i>TimeWindowStart</i> ), Hora de Fecho ( <i>TimeWindowEnd</i> )	<b>Outputs não espaciais:</b> Não tem
<b>Order Pairs</b>	<b>Inputs:</b> Identificador ( <i>objectoid</i> ), Primeira order do par a ser executada ( <i>FirstOrderName</i> ), Segunda Order a ser executada ( <i>SecondOrderName</i> ).	<b>Outputs:</b> Não tem

Tabela 3. Atributos dos Sub-layers do VRP

Fonte: adaptado do Help do ArcGIS

**Input:** Os pedidos já atribuídos aos vários transportadores e a informação de frota.

**Output:**

- Lista ordenada de paragens, hora de chegada à paragem, tipo de tarefa (recolher ou entregar);
- Visualização do/s percurso/s e descrição da rota (mudanças de direcção);
- Custo das viagens.

**Métodos:**

Através do método *AddLocation*, são criados todos os sub-layers atrás referidos, diferenciando alguns parâmetros consoante o tipo de viagem.

- planeamento semanal no caso do transportador TN, dos pedidos (nacionais) no período de uma semana, usando VRP no eixo lógico principal.

- planeamento diário no caso dos transportadores do tipo EP, dos pedidos (locais e regionais) de cada dia de cada EP, usando a Frota do EP, aplicando a metodologia VRP em rede viária.

Em ambos os casos pretende-se calcular a melhor forma de transportar todos os indivíduos aos seus destinos (cumprir todas as tarefas da lista) usando a frota disponível. ‘melhor forma’ significa em menos tempo/usando menos recursos, mas cumprindo as restrições horárias.

A seguir apresento a tabela 4 com os parâmetros principais que caracterizam e diferenciam as duas situações:

	<b>VRP1</b>	<b>VRP2</b>
utilização	VRP no eixo principal aplicado apenas ao transportador TN	VRP aplicado a nível local e regional para ser usado por todos os transportadores do tipo EP
Quantidade de delivery ou pickup	1 transportado	1 transportado
depots	depot inicial $\neq$ depot final (não coincidem)	depot inicial = depot final (coincidem)
routes	uma por carro da Frota do TN	uma por carro da frota do EP
tempo máximo de viagem	720 minutos (12 horas)	300 minutos (5 horas)
violação de tempo máxima permitida	1440 minutos (24 horas) Este parâmetro pode ser ajustado conforme as conveniências; quanto maior for mais partido se poderá tirar da lotação dos carros	5 minutos  Aqui as horas a comparecer nos locais (tribunais, hospitais), devem ser respeitadas.
order pairs	Apenas um par	Dois pares para incluir o regresso
	Os transportados entram num local e saem noutro onde ficam	Os transportados saem e regressam ao mesmo local depois de realizarem duas viagens
rede	Eixos lógicos, uma vez que o percurso é fixo e sempre igual, logo não há uma procura do caminho físico mas sim uma distribuição automática e que tem em conta a lotação e as janelas de tempo.	Rede física de estradas, pois há em cada dia uma rota diferente.

Tabela 4. Parâmetros de dois VRP

As figuras 20 e 21 mostram dois modelos da aplicação do VRP, um ilustrando a criação do layer de análise e outro um modelo geral incluindo dois VRP.

Figura 20. Modelo de criação de um VRP layer

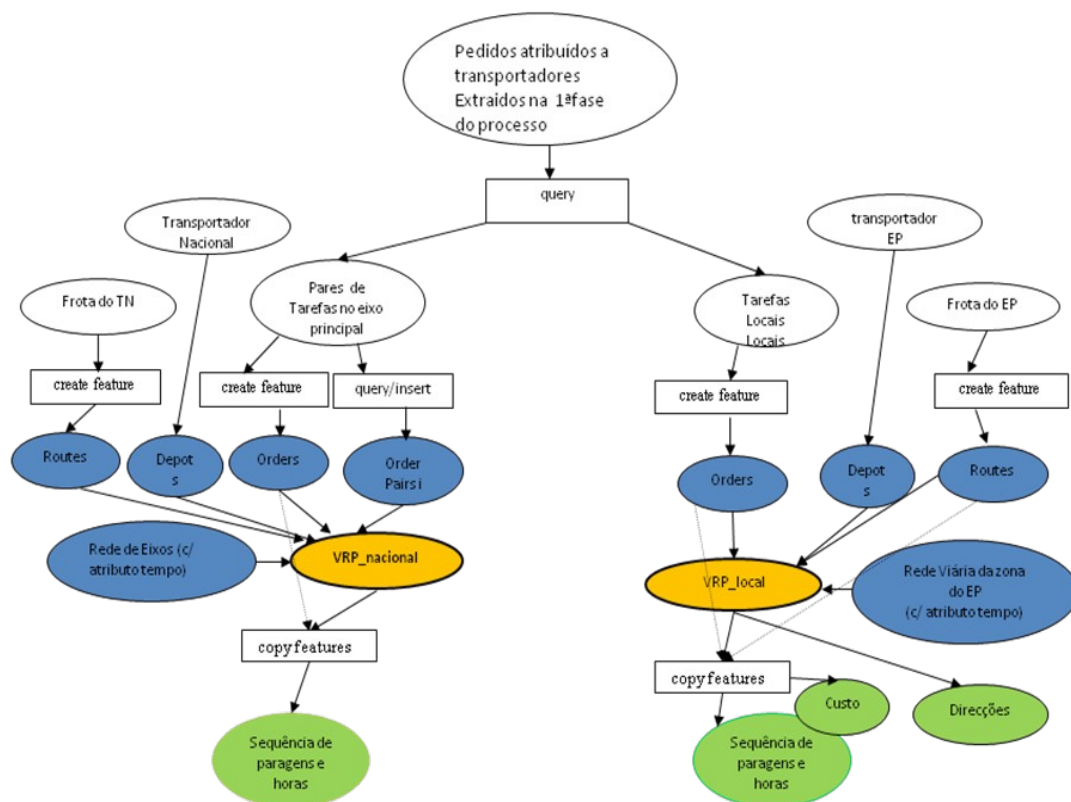


Figura 21. Modelo Geral de Planeamento de Viagens



### 3.3 Conclusões

Este capítulo descreveu a modelação geral do sistema, incluindo o estudo dos eixos necessários à construção de uma rede apropriada, e da obtenção dos dados necessários à sua construção. Seguiu-se a descrição das ferramentas necessárias à construção do sistema, e dos softwares e linguagens usadas.

O trabalho foi realizado, ora correndo o ficheiro Python, ora visualizando e inspeccionando ‘de perto’ os resultados através do ArcMap.

Constatou-se a relativa facilidade, embora trabalhosa, de construção de uma rede através do Network Analyst, permitindo criar facilmente as suas regras de conexão, e os seus sentidos.

A simplicidade da linguagem Python constituiu uma boa surpresa, assim como o facto de se poder gravar num *script* Python as funcionalidades criadas no *model builder*, que facilitaram bastante a compreensão de uma linguagem nova e principalmente na sua forma de chamar os *tools* do ArcGIS.

O ambiente Python facilitou bastante o trabalho de teste, pois que uma vez compreendidas as operações a aplicar à rede, o trabalho seguiu com a construção de um *script* em Python, que até permitiu a conjugação de algumas operações que não tinham sido conseguidas através do *model builder*, como por exemplo o uso do *sub-layer route*, do *layer Route* do VRP *layer*, na aplicação ‘Partição Pedidos’.

O ficheiro Python corre com suficiente rapidez (mais rápido o Python 2.5 que o Python 2.6), e o resultado obtido através do método *locate*, pode, ou não, incluir os campos associados a EP, o que facilita muito o trabalho, podendo-se variar, consoante se pretende visualizar e analisar os resultados, ou apenas transportar dados para um Excel.

A tarefa mais complicada a extracção dos eixos, que acabou por ser resolvida com a extracção de EP. No entanto, depois de conhecida a função ‘*locate*’, foi uma boa surpresa verificar que todos os pedidos de um ficheiro poderiam ser tratados de uma só vez, e portanto, a aplicação uma única vez de ‘Partição Pedidos’, transformava tantos pedidos quantos se quisesse nos seus vários sub-pedidos, não sendo necessário criar um ciclo que actuasse registo a registo, como inicialmente se julgou ser.

O projecto exigiu ainda programação em SQL e SQL/PL para criação de tabelas Oracle e procedimentos Oracle.

No caso da ‘Partição Pedido’, houve o objectivo de não ficar pelo básico, tentando responder também a situações mais complexas. Começou-se como uma rede simples contendo os eixos básicos, e depois foram-se acrescentado os eixos mais complicados da zona da grande Lisboa, e por fim, alguns eixos de tribunais.

Alguns eixos foram criados e depois abandonados, mas servindo no entanto para testar o funcionamento e eficácia desse tipo de eixos, assim como o trabalho investido. Foi o caso do eixos com ligações entre os EP da zona de Lisboa e tribunais, ou entre esses EP e EPAlcoentre e EPMontijo.

O raio de procura foi ajustado para 50000 metros, para que fossem localizados os tribunais a essa distância da rede.

A necessidade de ajuste do tipo de *snap*, foi uma descoberta importante, realizada a meio do trabalho, sem a qual não estaria a calcular o EP mais próximo do tribunal, mas sim o último EP de um caminho que terminava no ponto da rede mais próximo do destino (ver Figura 36), ponto esse que podia situar-se nos eixos, fora dos EP.

O prolongamento dos eixos até aos tribunais foi considerado dispensável, uma vez que a funcionalidade procura o evento (dentro dos *shapes* definidos no *tipo de snap*) da rede, mais próximo, dentro de um raio de 50000 metros. Assim determinou-se que o caminho terminaria nesse EP mais próximo.

Acabou por ser necessário incluir-se alguns desses eixos que fazem ligações a tribunais, precisamente quando não se pretende que vigore a regra ‘do EP mais próximo’.

Foi explorado o uso de usar ‘linear referencing’, em vez de coordenadas, nos locais de *input*. Isso produziria resultados mais rápidos pois usaria uma coordenada única, mas exigiria um primeiro trabalho de referenciar todos os locais de origem e destino, dentro da rede, que não seria compensador, uma vez que esta aplicação nunca lidará com muitos registos ao mesmo tempo e porque o tempo experimentado tanto no ArcGis, como no *script* Python são perfeitamente aceitáveis.

Os EP foram usados como nós de ligação pontos entre os eixos, apesar de serem dispensáveis, pois a rede ao ser construída cria as necessárias *junctions* entre os eixos que se tocam, se elas não existirem. A opção de os usar, foi aproveitar para testar o seu rigor em termos de coordenadas, ou seja comprovar que estão coincidentes com as extremidades dos eixos. Ao serem incluídos na rede o sistema avisa caso algum deles

fique desconectado da rede. Esta questão é importante porque eles são posteriormente usados para *locate* ou seja para serem intersectados com o caminho resultante, determinando o conjunto de EP que determinam o caminho.

Foi usada a janela direcções, para análise detalhada das mudanças de eixo, informação que se tentou mas que não se conseguiu extrair para um ficheiro de modo estruturado.

Depois de bem compreendido o funcionamento básico da rede e do *routing*, tentou-se resolver algumas das situações que caem fora do funcionamento básico que estava a ser implementado. Algumas dessas situações acrescentaram mais-valia à rede inicial. No entanto quando se tentou criar as ligações directas entre alguns EP dos eixos principais ao TN, primeira situação em que surge a possibilidade de mais do que um caminho, surgiu o desafio de não deixar que alguns caminhos enveredem pelo melhor caminho. Foram criados dois grupos de conectividade para poder restringir os seus pontos de conexão.

Apesar de o problema em questão ter sido resolvido, surgiram outros problemas, pois deixou de haver caminho entre algumas zonas e a solução encontrada foi criar duas redes para aplicar a situações diferentes.

Outras situações de caminhos não únicos, mas menos complicadas poder-se-ão implementar no pós-projecto, como sejam as ligações entre EP de uma mesma região, que não gerarão problemas desde que não pertençam ao eixo principal.

Os eixos lógicos funcionam e adaptam-se a situações específicas. A estrutura de eixos é fundamental e suficiente para determinar o comportamento da escolha do melhor caminho, que no caso da rede2 é único, mas não é o é no caso da rede3.

A aplicação do VRP ao planeamento de viagens nos eixos nacionais foi explorado o suficiente para comprovar que o problema é resolvido com a mesma rede de eixos lógicos, apenas usando a impedância ‘tempo’ em vez de ‘distância’, e também sem rede viária. Em relação à sua aplicação a viagens locais com recurso a redes viárias, foi apenas realizado um estudo teórico e comparativo.

Foram adquiridos conhecimentos variados, nomeadamente sobre construção de redes, aplicação de um *routing* e localização de eventos. E também nos softwares Network Analyst, ligação de *script* Python e a sua conjugação.

## 4. RESULTADOS

Em termos de resultados, apresenta-se, por um lado, o resultado directo do output da aplicação, por outro, uma avaliação concreta da utilidade da aplicação no contexto para o qual foi pensado.

Por um lado pretende-se comprovar que a aplicação executa de acordo com as regras previstas, apresentando os *inputs*, os *outputs* e análise da relação entre uns e outros.

A avaliação no contexto real, foi mais complicada pelo facto de não haver dados informatizados, tendo os poucos existentes sido informatizados expressamente para o efeito.

### 4.1 Preparação dos Dados

A partir da base de dados existente foram recolhidos 1181 pedidos iniciais, referentes a transportados que deveriam estar nos seus destinos entre 26 de Junho de 2011 e 26 de Julho de 2011, e devidamente tratados para poderem servir de input à funcionalidade ‘Partição Pedidos’, tanto directamente no ArcGIS como no *script* do ficheiro Python.

Depois de usados na ferramenta construída e produzidos os resultados, estes foram por um lado analisados no ArcGIS (visualização das *routes* e análise da informação contida nas janelas de direcção), no Excel depois de ordenados convenientemente, e seguidamente tratados em Oracle para produzir um resultado pronto a inserir nas tabelas da aplicação SIP usada diariamente na organização do caso de estudo, sob a forma de pedidos finais atribuídos a transportadores.

Foram também previamente realizadas alterações num dos ecrãs da aplicação existente, para se permitir inserir manualmente informação do TN referente aos pedidos reais em suporte de papel e foi inserida informação referente a uma viagem do TN em Junho de 2011, de modo a se poder posteriormente comparar com os resultados obtidos automaticamente, através das ferramentas construídas.

## 4.2 Resultados

### 4.2.1 Resultados de ‘Partição Pedidos’

Dos 1181 registos usados para testar a aplicação foram escolhidos alguns considerados significativos para a demonstração dos resultados. O critério foi abarcar diferentes combinações Origem–Destino. O que se pretende é comprovar que o caminho entre uma determinada origem e um determinado destino é o correcto dentro das regras atrás descritas.

A tabela 5 é apresentada, apenas para melhor compreensão do conteúdo das tabelas seguintes, contendo os nomes e os códigos, das entidades de origem e de destino, para cada ID.

PEDIDOS				
ID	COD ORIGEM	ORIGEM	COD DESTINO	DESTINO
30	1020	Est. Pris. de Coimbra	1253	Vara Competência Mista Cível e Criminal - 2ª Secção - Coimbra
37	1020	Est. Pris. de Coimbra	2514	4ª Vara Criminal de Lisboa
90	1030	Est. Pris. do Linho	1143	1º Juízo - Tribunal Judicial da Comarca de Beja
171	1040	Est. Pris. de Lisboa	1160	1º Juízo - Tribunal Judicial da Comarca de Beja
297	1060	Est. Pris. de Paços de Ferreira	1257	3º Juízo Criminal - Braga
301	1060	Est. Pris. de Paços de Ferreira	1419	4º Juízo Criminal - Coimbra
321	1070	Est. Pris. de Pinheiro da Cruz	1257	Ministério Público de Grândola
327	1070	Est. Pris. de Pinheiro da Cruz	1342	2º Juízo Competência Especializada Criminal - Faro
336	1070	Est. Pris. de Pinheiro da Cruz	2511	1ª Vara Criminal de Lisboa
616	1120	Est. Pris. de Leiria	1342	2º Juízo Competência Especializada Criminal - Faro
635	1120	Est. Pris. de Leiria	1284	1º Juízo - Tribunal Judicial da Comarca da Covilhã
766	1170	Est. Pris. de Izeda	2452	1º Juízo Competência Especializada Criminal - Viseu
847	2020	Est. Pris. Reg. de Aveiro	2526	Juízo de Instrução Criminal de Águeda
852	2030	Est. Pris. Reg. de Beja	2569	Juízo de Grande Instância Criminal de Sintra
974	2130	Est. Pris. Reg. da Guarda	2539	Juízo de Média Instância Criminal de Aveiro

Tabela 5. Tabela com Locais de Origem e Destino dos Pedidos

A tabela 6 mostra o conteúdo do ficheiro ‘pedidos\_escolhidos.xlsx’ (corrido na rede2), que é um dos *inputs* do *script* Python, e contém toda a informação necessária relativa a cada pedido.

FILE ID	TIPO	COD	COORDX	COORDY
30	1	1020	-8,418973	40,205944
30	2	1253	-8,4292014	40,211491
37	1	1020	-8,418973	40,205944
37	2	2514	-9,0973563	38,77061
90	1	1030	-9,384396	38,757089
90	2	1143	-8,6653745	37,716621
171	1	1040	-9,187106	38,749985
171	2	1143	-8,6653745	37,716621
297	1	1060	-8,440887	41,262031
297	2	1160	-8,4229034	41,551761
301	1	1060	-8,440887	41,262031
301	2	1257	-8,4292014	40,211491
321	1	1070	-8,752245	38,309083
321	2	1419	-8,561562	38,178692
327	1	1070	-8,752245	38,309083
327	2	1342	-7,9255812	37,017929
336	1	1070	-8,752245	38,309083
336	2	2511	-9,0973563	38,77061
616	1	1120	-8,806699	39,725796
616	2	1342	-7,9255812	37,017929
635	1	1120	-8,806699	39,725796
635	2	1284	-7,4990433	40,275579
766	1	1170	-6,7250358	41,564526
766	2	2452	-7,9162478	40,656331
847	1	2020	-8,657901	40,637518
847	2	2526	-8,4457254	40,573091
852	1	2030	-7,869228	38,018928
852	2	2569	-9,231466	38,729897
974	1	2130	-9,2554339	38,935502
974	2	2539	-8,6538139	40,63836

Tabela 6. Tabela de Pedidos Iniciais

A tabela 7 mostra o conteúdo do ficheiro de *output* ‘resultado\_escolhidos.dbf’, depois de gravado em Excel e ordenado para uma boa leitura do resultado. Este ficheiro foi gerado pelo *script* ‘ParticaoRede2.py’, e posteriormente ordenado pelas colunas FID e MEAS. FID é a identificação da *route* a que pertence cada evento detectado, e que corresponde também à identificação do pedido inicial, ou seja cada pedido inicial FID, formado por uma origem e um destino foi transformado numa sequência ordenada de EP com o mesmo FID, que constituem a *route* FID. MEAS é a distância do evento, ou seja do EP detectado à origem da *route* produzida, que nos permite dentro de cada *route* ordenar a sequência do princípio para o fim.

RID	MEAS	DESCRIÇÃO
30	0,000000000000	EPCoimbra
30	0,49329851012	EPRaveiro
37	0,000000000000	EPCoimbra
37	0,61131306779	EPRLeiria
37	0,62043795836	EPLeiria
37	1,13864873772	EPVJudeus
37	1,16272891486	EPAlcoentre
37	1,67488766580	EPLisboa
90	0,000000000000	EPLinho
90	0,19679137971	GISPl
90	0,41297089968	EPRMontijo
90	0,54732729476	EPRSetubal
90	0,89913286474	EPPCruz
90	1,82859988747	EPRBeja
171	0,000000000000	EPLisboa
171	0,01682410688	GISPl
171	0,23300362685	EPRMontijo
171	0,36736002193	EPRSetubal
171	0,71916559191	EPPCruz
171	1,64863261464	EPRBeja
297	0,000000000000	EPPFerreira
297	0,24156966119	EPPorto
297	0,67868753045	EPRBraga
301	0,000000000000	EPPFerreira
301	0,24156966119	EPPorto
301	0,80430541647	EPRaveiro

Tabela 7. Tabela de resultados (continua)

RID	MEAS	DESCRIÇÃO
327	0,000000000000	EPPCruz
327	0,92946702271	EPRBeja
327	1,93517834711	EPRSilves
327	2,46805230453	EPRFaro
336	0,000000000000	EPPCruz
336	0,35180556998	EPRSetubal
336	0,48616196506	EPRMontijo
336	0,70318515244	EPLisboa
616	0,000000000000	EPLeiria
616	0,51821077936	EPVJudeus
616	0,54229095650	EPAlcoentre
616	1,05444970744	EPLisboa
616	1,07127381432	GISPl
616	1,28745333428	EPRMontijo
616	1,42180972939	EPRSetubal
616	1,77361529934	EPPCruz
616	2,70308232207	EPRBeja
616	3,70879364645	EPRSilves
616	4,24166760387	EPRFaro
635	0,000000000000	EPLeiria
635	0,00912489057	EPRLeiria
635	0,62043795836	EPCoimbra
635	1,11373646848	EPRaveiro
635	2,32781576107	EPRCovilha
766	0,000000000000	EPIzeda
766	1,98314038327	EPPorto
766	2,54587613854	EPRaveiro
766	3,14619315114	EPRViseu
847	0,000000000000	EPRaveiro
852	0,000000000000	EPRBeja
852	0,92946702271	EPPCruz
852	1,28127259269	EPRSetubal
852	1,41562898777	EPRMontijo
852	1,63265217514	EPLisboa
974	0,000000000000	EPRGuarda
974	1,39367953126	EPRaveiro

Tabela 7. Tabela de resultados (continuação)



A tabela 8, mostra os resultados depois de alterados pelos procedimentos Oracle.

TRANSPORTADOR	ORIGEM	DESTINO	PEDIDO	ORDEM
GISP - 1º esquadrão	Est. Pris. de Coimbra	Est. Pris. Reg. de Aveiro	30	1
Est. Pris. Reg. de Aveiro	Est. Pris. Reg. de Aveiro	Vara Competência Mista Cível e Criminal - 2ª Secção - Coimbra	39	100
GISP - 1º esquadrão	Est. Pris. de Coimbra	Est. Pris. de Lisboa	37	1
Est. Pris. de Lisboa	Est. Pris. de Lisboa	4ª Vara Criminal de Lisboa	37	100
GISP - 1º esquadrão	Est. Pris. de Lisboa	Est. Pris. Reg. de Beja	171	1
Est. Pris. Reg. de Beja	Est. Pris. Reg. de Beja	1º Juízo - Tribunal Judicial da Comarca de Beja	171	100
Est. Pris. de Paços de Ferreira	Est. Pris. de Paços de Ferreira	Est. Pris. do Porto	297	1
Est. Pris. do Porto	Est. Pris. do Porto	Est. Pris. Reg. Braga	297	2
Est. Pris. Reg. Braga	Est. Pris. Reg. Braga	3º Juízo Criminal - Braga	297	100
Est. Pris. de Paços de Ferreira	Est. Pris. de Paços de Ferreira	Est. Pris. do Porto	301	1
GISP - 1º esquadrão	Est. Pris. do Porto	Est. Pris. Reg. de Aveiro	301	2
Est. Pris. Reg. de Aveiro	Est. Pris. Reg. de Aveiro	4º Juízo Criminal - Coimbra	301	100
GISP - 1º esquadrão	Est. Pris. de Pinheiro da Cruz	Est. Pris. Reg. de Faro	327	1
Est. Pris. Reg. de Faro	Est. Pris. Reg. de Faro	2º Juízo Competência Especializada Criminal - Faro	327	100
GISP - 1º esquadrão	Est. Pris. de Pinheiro da Cruz	Est. Pris. de Lisboa	336	1
Est. Pris. de Lisboa	Est. Pris. de Lisboa	1ª Vara Criminal de Lisboa	336	100
GISP - 1º esquadrão	Est. Pris. de Leiria	Est. Pris. de Lisboa	616	1
Est. Pris. de Lisboa	Est. Pris. de Lisboa	GISP - 1º esquadrão	616	2
GISP - 1º esquadrão	GISP - 1º esquadrão	Est. Pris. Reg. de Faro	616	3
Est. Pris. Reg. de Faro	Est. Pris. Reg. de Faro	2º Juízo Competência Especializada Criminal - Faro	616	100
GISP - 1º esquadrão	Est. Pris. de Leiria	Est. Pris. Reg. de Aveiro	635	1
Est. Pris. Reg. de Aveiro	Est. Pris. Reg. de Aveiro	Est. Pris. Reg. da Covilhã	635	2
Est. Pris. Reg. da Covilhã	Est. Pris. Reg. da Covilhã	1º Juízo - Tribunal Judicial da Comarca da Covilhã	635	100
Est. Pris. de Izeda	Est. Pris. de Izeda	Est. Pris. do Porto	766	1
GISP - 1º esquadrão	Est. Pris. do Porto	Est. Pris. Reg. de Aveiro	766	2
Est. Pris. Reg. de Viseu	Est. Pris. Reg. de Aveiro	Est. Pris. Reg. de Viseu	766	3
Est. Pris. Reg. de Viseu	Est. Pris. Reg. de Viseu	1º Juízo Competência Especializada Criminal - Viseu	766	4
Est. Pris. Reg. de Aveiro	Est. Pris. Reg. de Aveiro	Juízo de Instrução Criminal de Águeda	847	100
GISP - 1º esquadrão	Est. Pris. Reg. de Beja	Est. Pris. de Lisboa	852	1
Est. Pris. de Lisboa	Est. Pris. de Lisboa	Juízo de Grande Instância Criminal de Sintra	852	100
Est. Pris. Reg. de Guarda	Est. Pris. Reg. de Guarda	Est. Pris. Reg. de Aveiro	974	1
Est. Pris. Reg. de Aveiro	Est. Pris. Reg. de Aveiro	Juízo de Média Instância Criminal de Aveiro	974	100

Tabela 8. Tabela de Resultados depois de alterados pelos procedimentos Oracle

Pegando e analisando alguns casos que ligam diferentes regiões do país, começa-se por apresentar casos tratados pela ‘Rede2’:

#### 1º Caso - RID 635 - EPLeiria -> tribunal da Covilhã

Existe um pedido de transporte de EPLeiria que pertence ao eixo nacional e fica situado em Leiria, para um tribunal localizado na Covilhã. O EP mais próximo deste tribunal é EPCovilhã. Então, o resultado será o caminho entre EPLeiria e EPCovilhã, sob a forma de uma lista sequencial de EP. Na ‘route’ gerada, representada graficamente pela figura 22, o resultado é:

EPLeiria, EPRLeiria, EPCoimbra, EPAveiro, EPCovilhã, que depois de sujeito ao procedimentos Oracle dará origem a duas viagens realizadas pelos transportadores TN: EPLeiria -> EPAveiro e EPAveiro: EPAveiro-> EPCovilhã e por fim a viagem local EPCovilhã-> tribunal da Covilhã, pelo transportador EPCovilhã.

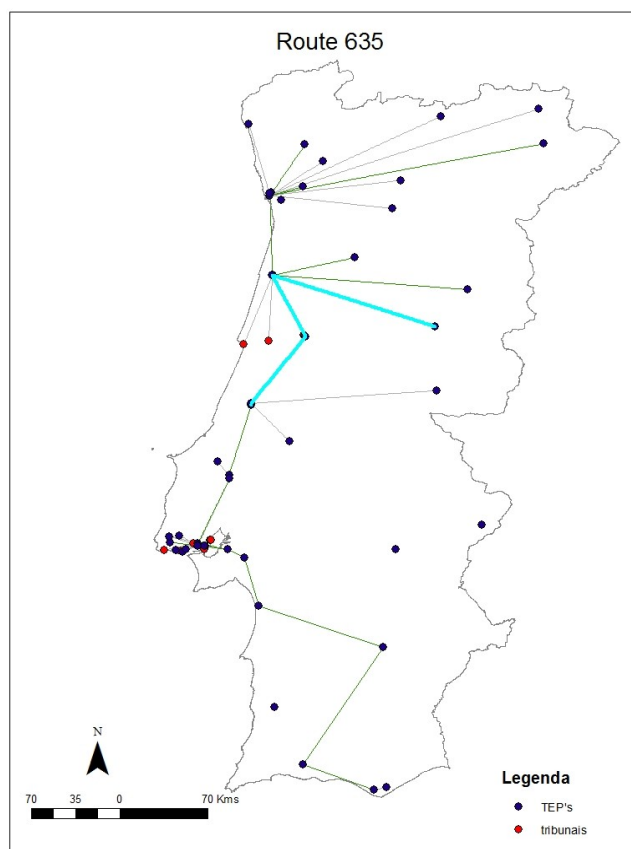


Figura 22. Caminho 635: entre EPLeiria e um tribunal da Covilhã

## 2º Caso - 766 EPIzeda -> tribunal de Viseu (figura 23)

Existe um pedido de transporte de EPIzeda que pertence a um eixo regional ligado ao TN no EPPorto, para um tribunal localizado em Viseu. O EP mais próximo deste tribunal é EPViseu. Então vai ser calculado o caminho entre EPIzeda e EPViseu, sendo gerada a sequência, EPIzeda, EPPorto, EPAveiro, EPViseu, dando origem a 3 viagens realizadas pelos transportadores EPIzeda, TN e EPAveiro, e por fim a viagem ao tribunal por EPViseu.

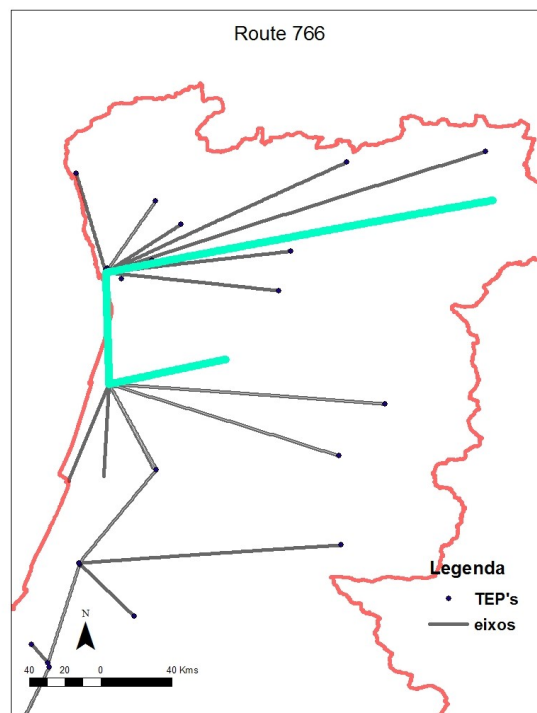


Figura 23. Caminho 766: entre EPIzeda e um tribunal de Viseu

## 3º Caso - 37- EPCoimbra -> tribunal de Lisboa (figura 24)

. Há um pedido de transporte do EPCoimbra que pertence ao eixo nacional para um tribunal localizado em Lisboa. O EP mais próximo deste tribunal é EPLisboa. Então vai ser calculado o caminho entre EPCoimbra e EPLisboa, sendo gerada a seguinte sequência: EPCoimbra,, EPRleiria, EPLeiria, EPVjudeus, EPAlcoentre, EPLisboa, dando origem a apenas uma viagem realizada pelo transportador TN, seguida da viagem até ao tribunal. Este caso poderá ter que ser alterado manualmente, se no contexto real o 'EP escolhido', não for o EPLisboa.



#### 5º Caso - 852 EPBeja-> tribunal de Sintra (figura 26)

Existe um pedido de transporte de EPBeja localizado em Beja, para um tribunal localizado em Sintra. O EP mais próximo deste tribunal é EPSintra, no entanto como pertence à zona da Grande Lisboa, será considerado o EPLisboa uma vez que se optou por não usar aqui a regra do EP mais próximo. EPLisboa poderá ser posteriormente substituído por qualquer um dos EP da zona de Lisboa.

Para já vai ser calculado o caminho entre EPBeja e EPLisboa, sendo gerada a seguinte sequência: EPBeja, EPPCruz, EPPsetubal, EPMontijo, EPLisboa, que depois de sujeitos ao procedimentos Oracle dará origem a 1 viagem, realizada pelo TN: EPBeja, EPLisboa, seguida da ida a tribunal. Este é outro caso que pode ter que ser alterado manualmente, caso o ‘EP escolhido’, na realidade não seja o EPLisboa.

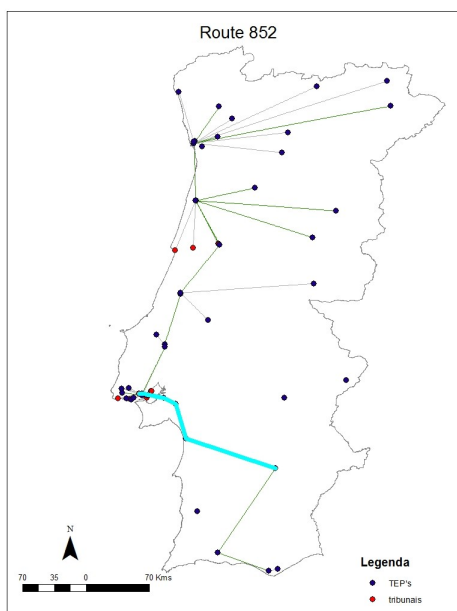


Figura 26. Caminho 852: entre EPBeja e um tribunal de Sintra.

#### 6º Caso – ID 327 - EPPcruz ->tribunal de Faro (figura 27)

Existe um pedido de transporte de EPPCruz que pertence ao eixo nacional sul, para um tribunal localizado em Faro. O EP mais próximo deste tribunal é EPFaro Então vai ser calculado o caminho entre EPPCruz e EPFaro, sendo gerada a seguinte sequência: EPPCruz, EPBeja, EPSilves, EPFaro, que irá gerar um único transporte realizado pelo transportador TN: EPPCruz->EPFaro, seguido da ida a tribunal pelo EPFaro.

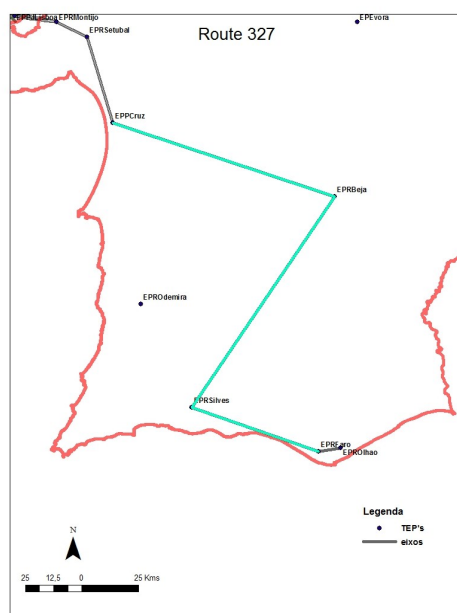


Figura 27. Caminho 327: entre EPPCruz e um tribunal de Faro

#### 7º Caso - 847 EPAveiro-> tribunal de Águeda (figura 28)

Existe um pedido de transporte entre EPAveiro e um tribunal localizado em Águeda. O EP mais próximo deste tribunal é o próprio EPAveiro único nó da rede incluído no caminho. Será gerada uma viagem local entre EPAveiro e o tribunal de Águeda.

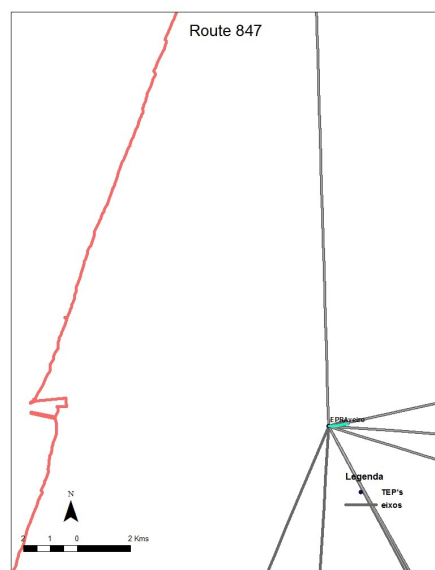


Figura 28. Caminho 847: entre EPAveiro e um tribunal de Águeda.

8º Caso - 974 EPGuarda -> tribunal de Aveiro (figura 29)

Existe um pedido de transporte entre EPGuarda que pertence ao eixo centro e um tribunal localizado em Aveiro. O EP mais próximo deste tribunal é EPAveiro então vai ser calculado o caminho entre EPGuarda e EPAveiro, sendo gerada a seguinte sequência: EPGuarda, EPAveiro, gerando uma única viagem realizada pelo EPGuarda, EPGuarda-> EPAveiro, seguida da viagem local realizada pelo EPAveiro, EPAveiro-> tribunal de Aveiro.

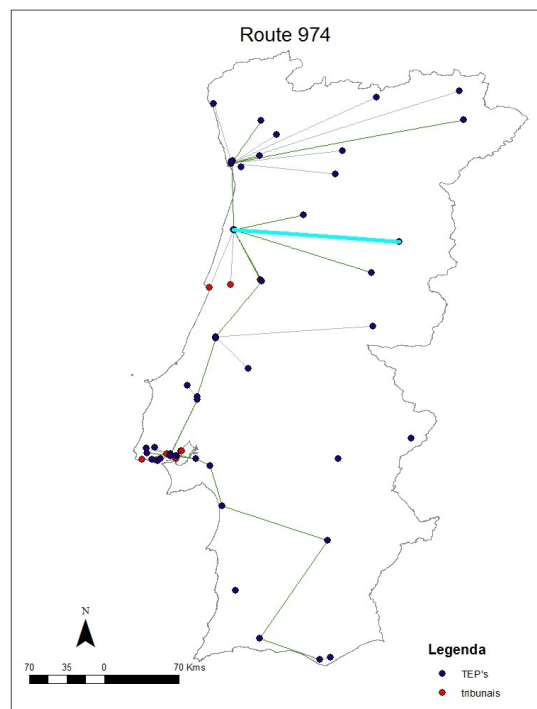


Figura 29. Caminho 974: entre EPGuarda e um tribunal de Aveiro

9º Caso – 297 EPPFerreira – tribunal de Braga (figura 30)

Existe um pedido de transporte entre EPPFerreira situado em Paços de Ferreira e um tribunal localizado em Braga. O EP mais próximo deste tribunal é EPBraga. Então vai ser calculado o caminho entre EPPFerreira e EPBraga, sendo gerada a seguinte sequência: EPPFerreira, EPPorto, EPBraga, gerando duas viagens, para além da ida a tribunal.

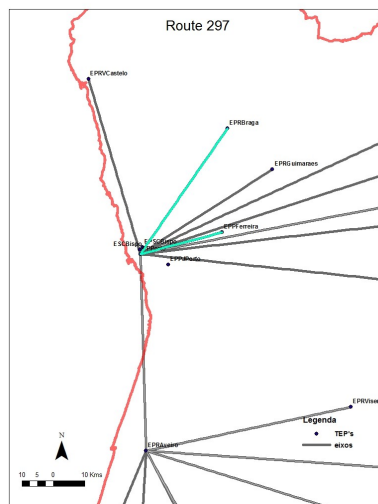


Figura 30. Caminho 297: entre EPPFerreira e um tribunal de Braga.

#### 10º Caso - 321 EPPCruz->tr Grândola

Este caso é aqui referido porque na primeira aplicação de *routing* não era encontrado, pois o raio era muito pequeno (resolvido depois de ser aumentado o raio). Resulta numa única viagem entre, EPPcruz e o tribunal de Grândola, e portanto não origina uma *route*.

11º Caso - 301 EPPferreira->tr Coimbra (figura 31) (corrigido da rede1 c/ o eixo tribunais de Aveiro). É gerada a sequência de EPPferreira, EPPorto, EPAveiro. Este último leva ao tribunal.

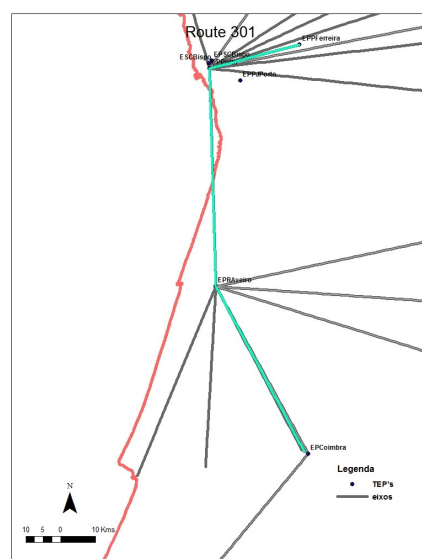




Figura 31. Caminho 301: entre EPPferreira e um tribunal de Coimbra.

12º Caso - 30 EPCoimbra-> tr Coimbra (figura 32) Ficou incorrecto na ‘Rede2’, devido ao Eixo\_tribunais\_Aveiro; pois trata-se apenas de uma viagem local. Ficaria resolvido se os casos em que o ‘EP origem’, é o mais próximo do tribunal, fossem previamente removidos.

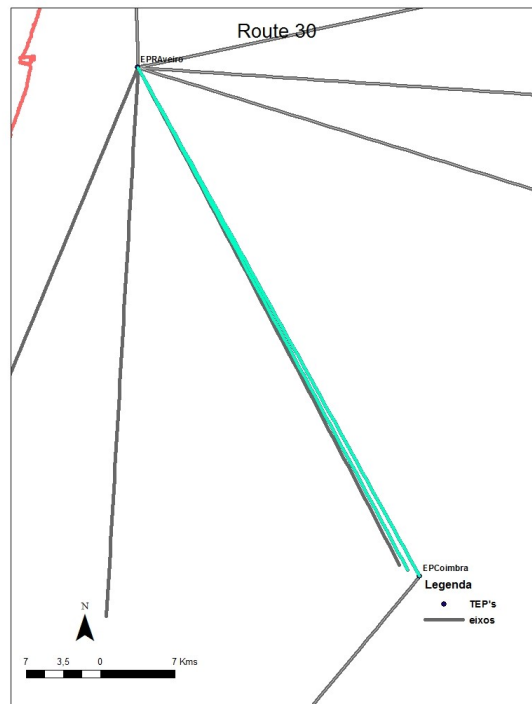


Figura 32. Caminho 30: entre EPCoimbra e um tribunal de Coimbra.

13º Caso - 616 EPLeiria-> trFaro (figura 33) (para ser corrigido pela ‘Rede3’)

Existe um pedido de transporte de EPLeiria que pertence ao eixo nacional norte, para um tribunal localizado em Faro. O EP mais próximo deste tribunal é EPFaro Então, vai ser calculado o caminho entre EPLeiria e EPFaro, sendo gerada a seguinte sequência: EPLeiria, EPVjudeus, EPA Alcoentre, EPLisboa, TN, EPMontijo, EPSetubal, EPPCruz, EPBeja, EPSilves, EPFaro, que irá gerar 3 viagens: uma realizada pelo TN: EPLeiria->EPLisboa, outra pelo EPLisboa: EPLisboa->TN e outra realizada pelo TN: TN-> EPFaro , e finalmente a ida a tribunal pelo EPFaro.

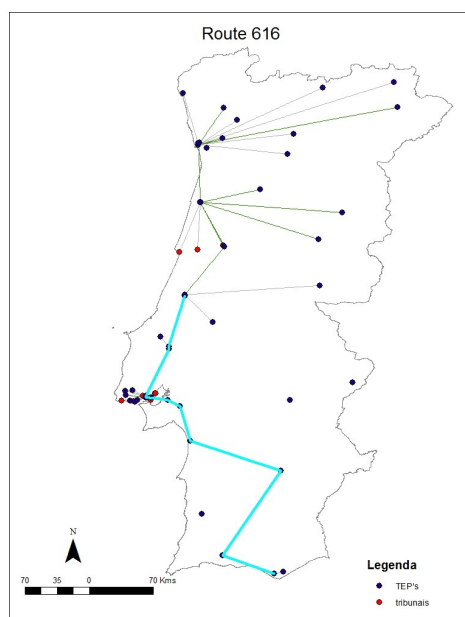


Figura 33. Caminho 616: entre EPLeiria e um tribunal de Faro.

14º Caso - 336 EPPcruz → tribunal de Lisboa (figura 34): Neste caso, na *route* criada está visível também o eixo do tribunal, embora o tribunal não apareça na sequência de nós da rede, pois o *locate* apenas faz a intersecção com os EP.

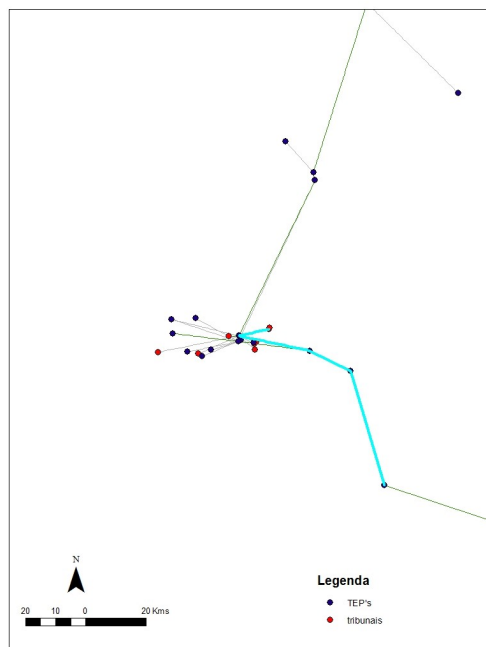


Figura 34. Caminho 336: entre EPPCruz e um tribunal de Lisboa.

16º Caso 616 EPLeia->trFaro (figura 35) (13º caso corrigido pela ‘Rede3’)

Existe um pedido de transporte de EPLeia que pertence ao eixo nacional norte, para um tribunal localizado em Faro. O EP mais próximo deste tribunal é EPFaro Então vai ser calculado o caminho entre EPLeia e EPFaro, sendo gerada a seguinte sequência: EPLeia, sedeTN, EPMontijo, EPSetubal, EPPcruz, EPBeja, EPSilves, EPFaro, que irá gerar duas viagens, uma realizada pelo EPLeia, EPLeia->sedeTN e outra pelo TN, TN-> EPFaro, seguida da ida tribunal.

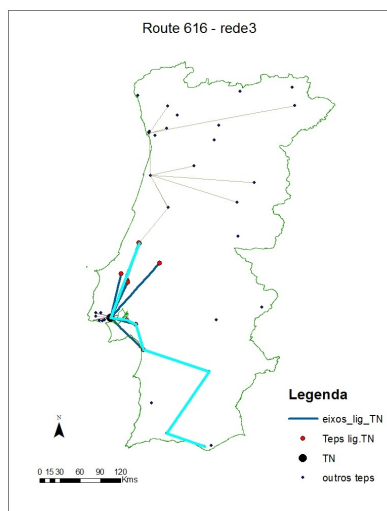


Figura 35. Caminho 616: entre EPLeia e um tribunal de Faro –‘Rede3’.

A comparação destes resultados, com os dados reais, começou por ser realizada com dados obtidos através da rede1. Foram comparados com os pedidos inseridos manualmente respeitantes a apenas duas viagens realizadas pelo TN, nos sentidos Lisboa->Porto e Porto->Lisboa em 21, 22 de Junho de 2011.

Numa primeira comparação foram em número reduzido os registos que coincidiram com as tarefas realizadas pelo TN. Depois de serem analisados um a um esses os registos, detectaram-se os motivos pelos quais houve registos reais sem correspondência nos resultados:

- Situações não contempladas por não serem idas a tribunal.
- O pedido não tinha sido contemplado na lista, apesar de se encontrar dentro do período previsto – depois de analisada a situação concluiu-se que nas passagens de Oracle para Excel houve alguns pedidos que ficaram de fora.

- Pedidos para tribunais de Coimbra, nos resultados davam viagens até ao EP<sub>Coimbra</sub>, enquanto na realidade tinham ido até ao EP<sub>Aveiro</sub> (por neste caso não ser usada a regra do ‘EP mais próximo’).
- Algumas diferenças entre entregas no EP<sub>Leiria</sub> e no EP<sub>RLeiria</sub>, que são EP muito próximos.

Também foram detectados os motivos para que os resultados obtidos não tenham correspondência nos registos reais:

- A ida a tribunal acabou por não se realizar.
- Foi directo (por ser próximo).
- Foi directo (apesar de não muito próximo; exemplo: Coimbra-Lisboa).
- Viagem Local (zona de Lisboa).
- Pedido errado (EP de origem incorrecto).
- Pedidos entretanto apagados.
- O transportado tinha entretanto sido internado no hospital.
- Tinha entretanto mudado de EP.
- Correspondem a viagens no eixo sul.
- Tarefa realizada numa viagem posterior (semanas seguintes).

Forma de resolver alguns desses problemas:

- Resolver a questão de Coimbra-Aveiro que é uma excepção à regra do ‘EP mais próximo’.
- Criar as ligações directas ao TN, a partir de alguns EP.
- Inserir pedidos das viagens para sul e viagens seguintes.

A resolução de alguns destes problemas deu origem à rede<sub>2</sub>, através da qual cerca de 70% dos pedidos reais coincidiam com os registos do ficheiro de output.

#### **4.2.1.1 Outras Questões**

Algumas situações ao nível da parametrização que só foram percebidas durante a exploração de resultados.

Tipo de SNAP – foi alterado o *tipo de snap* inicialmente para toda a rede (figura 36) de modo a ser apenas para os EP, para que os caminhos terminem exactamente num EP. Este parâmetro é passado através do método *AddLocation*.

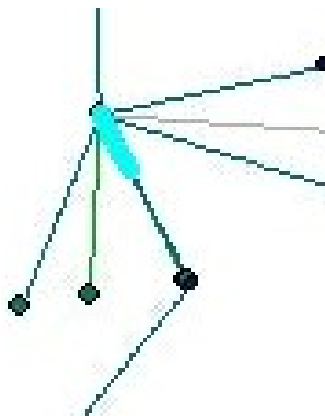


Figura 36. Ilustração de um tipo de *snap* para toda a rede

RAIO – um raio de procura menor do que a distância entre o local de destino e o EP mais próximo produz no ArcMap a mensagem ‘*do not locate*’.

O resultado da 1ª experiência usando a rede1 produziu um ficheiro com poucos registos. Este problema foi resolvido com o aumento do raio de procura, que passou a ser 50000, resolvendo o problema.

#### 4.2.2 Resultados Planeamento de Viagem

Foi testado o ficheiro ‘vrp\_gisp.py’, ficheiro Python que contendo a sequência de funcionalidades do ArcGIS para construir e executar um layer VRP. Foi usado um input de 5 pedidos.

**Depots (input)** – Sendo este um tipo de VRP em que as viagens começam num local e acabam noutro, são passadas as coordenadas do inicio e do fim da viagem, logo existe mais do que um depósito (tabela 9)

OBJECTID	NAME	DEPOTX	DEPOTY
1	GispLisboa	-9,187106	38,74999
2	GispPorto	-8,67436	41,20001

Tabela 9. Tabela de Depots (Depósitos) do VRP

**Orders (input)** – Como se vê na tabela 10, são passadas as coordenadas do local de cada tarefa, e a data limite para a sua execução. Sendo este um VRP com *pickup-e-delivery*, ou seja, em que os transportados são levados de um local para outro, o ‘levantamento’ do transportado será uma tarefa (pickup=1 e delivery=0) e a ‘entrega’ será outra tarefa (pickup=0 e delivery=1), será outra. A relação entre um ‘levantamento’ e uma ‘entrega’ será estabelecida na tabela 11.

ID		DATA-				DELI	PICK
PEDIDO	ORDERS	LIMITE	EP	COORDX	COORDY	VERY	UP
18	18_1_1010	23-06-2011	1010	-8,95958	39,20883	1	0
18	18_1_1120	23-06-2011	1120	-8,8067	39,7258	0	1
19	19_1_1010	05-07-2011	1010	-8,95958	39,20883	1	0
19	19_1_1120	05-07-2011	1120	-8,8067	39,7258	0	1
59	59_1_1020	29-06-2011	1020	-8,41897	40,20594	1	0
59	59_1_1080	29-06-2011	1080	-8,67436	41,20001	0	1
80	80_2_1040	23-06-2011	1040	-9,18711	38,74999	1	0
80	80_2_1120	23-06-2011	1120	-8,8067	39,7258	0	1
122	122_1_1040	30-06-2011	1040	-9,18711	38,74999	1	0
122	122_1_1080	30-06-2011	1080	-8,67436	41,20001	0	1

Tabela 10. Tabela de input de *Orders* (Tarefas) do VRP

**Order Pairs (input)**- Sendo este um VRP com *pickup-e-delivery* foi passada informação que emparelha as tarefas (tabela 11)

OBJECTID	FIRST ORDER	SECOND ORDER	DESCRIÇÃO
1	59_1_1020	59_1_1080	Pedido 59 de EPCoimbra para EPPorto
2	18_1_1010	18_1_1120	Pedido 18 de EPAlcoentre para EPLeiria
3	19_1_1010	19_1_1120	Pedido 19 de EPAlcoentre para EPLeiria
4	80_2_1040	80_2_1120	Pedido 80 de EPLisboa para EPLeiria
5	122_1_1040	122_1_1080	Pedido 122 de EPLisboa para EPPorto

Tabela 11. Tabela de Order Pairs do VRP

**Routes (input)** – Contem informação relativa a cada viagem/carro (tabela 12).

ID	NAME	START DEPOT	END DEPOT	CAPACITIES	MAX ORDER	START TIME	END TIME
1	LP21062011A	GispLisboa	GispPorto	16	30	21-08-2011	21-08-2011
2	LP21062011B	GispLisboa	GispPorto	16	30	21-08-2011	21-08-2011
3	PL22062011A	GispPorto	GispLisboa	16	30	22-08-2011	22-08-2011
4	PL22062011B	GispPorto	GispLisboa	16	30	22-08-2011	22-08-2011

Tabela 12. Tabela de input de Routes do VRP

**Orders (output)** – O *solve*, que executa o VRP, cria algumas colunas, com a informação que pretendemos: *route\_name*, sequência, *cumul\_travel\_time*, *arrive\_time*, e *depart\_time*. Esta informação associa tarefas a viagens, os pedidos a viagens e coloca-os por ordem dentro da viagem, gerando o planeamento que se pretende (tabela 13). Esta tabela devidamente associada coma a tabela 9, permite-nos também saber que tarefas realizar em cada nó do percurso.

Orders	Route Name	Sequence	Travel Time	Arrive Time	Depart Time	Delivery qt.	Pickup Qt.
122_1_1040	LP21062011A	2	0,0000	21-06-2011	21-06-2011	0	1
80_2_1040	LP21062011A	3	0,0000	21-06-2011	21-06-2011	0	1
18_1_1010	LP21062011A	4	90,0000	21-06-2011	21-06-2011	0	1
19_1_1010	LP21062011A	5	90,0000	21-06-2011	21-06-2011	0	1
80_2_1120	LP21062011A	6	180,0000	21-06-2011	21-06-2011	1	0
19_1_1120	LP21062011A	7	180,0000	21-06-2011	21-06-2011	1	0
18_1_1120	LP21062011A	8	180,0000	21-06-2011	21-06-2011	1	0
59_1_1020	LP21062011A	9	270,0000	21-06-2011	21-06-2011	0	1
59_1_1080	LP21062011A	10	450,0000	21-06-2011	21-06-2011	1	0
122_1_1080	LP21062011A	11	450,0000	21-06-2011	21-06-2011	1	0

Tabela 13. Tabela de output de Orders do VRP

**Routes (output)** - Imagem gráfica da *route* resultante (figura 37). A *route* não é o mais importante neste problema, mas sim as atribuições de ‘recolha’ ou ‘entrega’, em cada nó, informação contida na tabela 13.

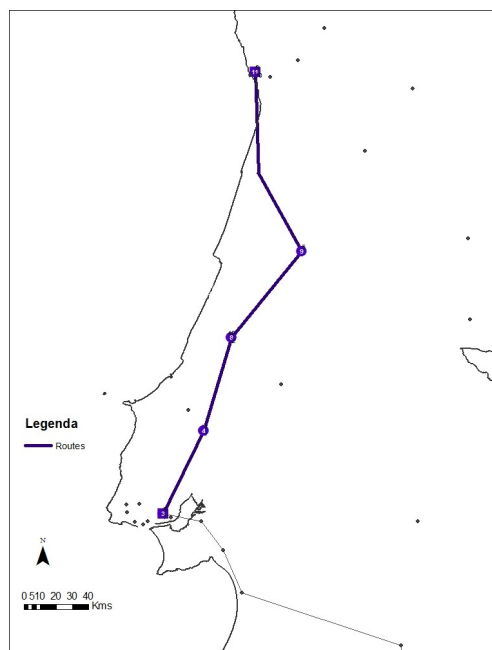


Figura 37. Percurso e paragens da route LP21062011

Neste caso, como se pode ver na tabela *Orders*, o resultado foi realizar todas as tarefas numa única viagem, a viagem de dia 21 no sentido Lisboa-Porto, uma vez que:

- a lotação de um carro o permitia
- as janelas de tempo apenas têm limites máximas, todos superiores ao dia da viagem
- todas as tarefas são no sentido Lisboa->Porto

Foram realizadas outras experiências com poucos pedidos, que resultaram em mais do que uma viagem. Por exemplo, ao ser incluída uma tarefa (ou melhor um par de tarefas) com a relação recolha->entrega no sentido Porto->Lisboa, essa tarefa foi associada à viagem PL22062011A, realizada no sentido Porto->Lisboa no dia 22.



## 5 – CONCLUSÃO

### 5.1 Avaliação do Projecto

Inicialmente focado na questão do planeamento de viagens, este trabalho acabou por se virar bastante mais para a questão da Partição de Pedidos, por um lado por ser necessária e anterior ao planeamento de viagens, por outro, por ser muito útil em termos práticos, no dia-a-dia da organização, e finalmente, por se ter tornado um desafio conseguir construir uma ferramenta que concretizasse exactamente o que se pretende.

Considereei interessante o facto de aplicar a eixos lógicos, funcionalidades que normalmente são aplicadas a eixos físicos.

É bom o facto de ser possível e fácil adaptar a construção da rede a situações muito específicas, significando que no futuro poderá adaptar-se a novas excepções.

Em relação à conexão directa com o Oracle acabou-se por optar em não o fazer nesta fase, uma vez que a rede informática teve muitos problemas de lentidão e não era o momento correcto para incluir novas ligações.

Por outro lado, estando prevista uma futura evolução para Oracle11g, permitindo a passagem directa de um ficheiro XML para uma tabela Oracle permitirá uma exploração mais interessante, uma vez que a ferramenta gera automaticamente um ficheiro XML.

Mesmo assim, o processo usado é relativamente simples, e sê-lo-ia muito mais tendo o ArcGIS instalado na máquina que acede ao Oracle.

Em relação ao facto de ter realizado a georreferenciação de tribunais (assim como dos estabelecimentos prisionais) manualmente, local a local, quando poderia ter usado trabalho realizado por outros, ou aplicar um *geocoding* através das moradas, deveu-se a considerar que acabaria por ser mais pratico e uma vez que é trabalho apenas realizado uma vez, pois é uma lista relativamente estável, e no caso do tribunais apenas ser necessário geo-referenciar os tribunais da amostra de pedidos, ou seja, cerca de 167 tribunais, e assim aproveitar para conhecer e testar um software de *geocoding* acessível a todos através da internet.

A criação das redes teve como objectivo, não só a concretização de uma ferramenta útil neste momento, como a exploração das potencialidades do método usado para responder a todas as particularidades de um serviço, pouco rígido na sua essência, e pouco estável no momento actual, uma vez que todos os organismos do

Estado se encontram em reestruturação e a procurar meios mais eficientes.

Portanto, ao terem sido experimentados novos eixos e alguns tendo acabado por ser postos de parte, não foi tanto para responder a situações que aconteçam muitas vezes, mas para explorar a possibilidade de sair fora do caso geral, uma vez que a realidade do contexto em que se insere o caso estudado é, como se disse, inerentemente instável, pelo que um bom auxiliar de gestão e de planeamento terá que ter capacidade de evoluir e de responder a um conjunto complexo de regras.

Por essa razão, a hipótese de usar a rede3 separadamente para os casos específicos, para a qual foi criada, sugere acima de tudo que haverá sempre a possibilidade de resolver um problema desta forma, sem que seja posta de parte a possibilidade de o conseguir de outro modo, por exemplo incorporando mais programação e criando redes mais complicadas.

Em relação à funcionalidade Planeamento de Viagens, apesar de não ter sido explorada, com muitos registos, ficou demonstrado que poderá ser uma boa ferramenta de agendamento, cumprindo a sua função de *scheduling*, por neste caso particular não se colocar a questão do *routing*, pois a sequência dos nós já está previamente estabelecida. No entanto, a existência de pares de tarefas, levantamento-entrega, torna esta tarefa especialmente difícil e portanto a criação automática de um *scheduling* eficiente, torna-se especialmente útil.

Resumindo, os aspectos positivos serão:

- A utilidade;
- A simplicidade das redes;
- A flexibilidade das redes;
- A simplicidade e rapidez do ficheiro Python, e o facto de incluir funcionalidades geográficas e não geográficas;
- As potencialidades do ArcGIS;
- A visualização durante os testes;
- A visualização dos resultados.

E os negativos serão:

- O uso dos ficheiros Excel;
- Não haver uma extracção directa de eixos.

## Conclusões

As Vantagens do SIG-T são óbvias, embora este projecto tenha usado uma ínfima parte das suas potencialidades. A criação de redes, a aplicação das operações de optimização, e a visualização de todo o processo, permite a criação de aplicações adaptáveis às necessidades reais.

Conclui-se que os Sistemas de Informação Geográfica são uma enorme mais-valia na resolução dos problemas de transporte, aplicáveis a um sem-número de situações diversas e que existe software suficientemente avançado para a criação de aplicações complexas adaptadas à realidade.

### 5.3 Desenvolvimentos Futuros

Num futuro próximo, pretende-se uma gradual exploração das potencialidades do projecto, para resolver todas as necessidades e ambições dos utilizadores para os quais foi concebido:

- Utilização das ferramentas criadas neste projecto, aplicadas periodicamente a grupos de registos, seguidas das alterações necessárias.
- Integração de ArcGIS/Python/Oracle tudo na mesma máquina.
- Disponibilização dos mapas obtidos, integrando-os na aplicação Oracle.
- Uso da ferramenta para simulação/avaliação de novos eixos.
- Conseguir num só ficheiro Python, aceder à Base de dados Oracle, aplicar as funcionalidades ArcGIS a essa informação, passar para o Oracle a informação resultante contida no XML gerado, e aplicar os restantes procedimentos Oracle.
- Criar um ficheiro Python com as funcionalidades de criação de rede, de preferência acedendo à base de dados *Oracle* para extrair a informação dos eixos, facilitando a alteração da rede ou criação de novas redes para simulação.
- Extrair a informação de ‘mudança de eixo’ que é mostrada na janela de direcções.
- Continuar a explorar a linguagem Python explorando a sua capacidade de ligar diferentes ambientes.

## REFERÊNCIAS BIBLIOGRÁFICAS

BARRICO, C, 1998, *Uma Abordagem ao Problema de Caminho Mais Curto Multiobjectivo – Aplicação ao Problema de Encaminhamento em Redes Integradas de Comunicações*. Dissertação de Mestrado em Sistemas e Automação / Área de Gestão da Informação.

DANTZIG, G. B. and Ramser J.H., 1959 The Truck Dispatching Problem, *Management Science* 6, 80-91.

FILHO,G., ARARAKI,R., AQUINO,M., Lorena,L., 2007, TerraNetwork: Sistema de Análise de Redes Urbanas, *In Anais XIII Simpósio Brasileiro de Sensoriamento Remoto*, (Florianópolis, INPE), 347-3055 (URL.:<http://www.lac.inpe.br/~lorena/terranet/terranetworksbsr.pdf>, consulta em 6-11-2009).

GOODCHILD, M., 2000, GIS and Transportation: Status and Challenges. *GeoInformatica* 4:2, 127-139, (Kluwer Academic Publishers).

HONEYCUTT, D., WYNNE, D., 2010, Getting Started With Python In ArcGIS, *in Esri Developer Summit*, 22-25 Março, Palm springs, CA.

LAPORTE, G., 1995, Routing Problems: A bibliography, *In Annals of Operations Research* 61, 227-262.

LORENA, L., NARCISO, M., 1999, ARSIG para o Domínio Agropecuário. *Agrosoft 99, II Congresso da SBI-Agro* (URL.: [www.agrosoft.org.br/trabalhos/ag99/artigo06.htm](http://www.agrosoft.org.br/trabalhos/ag99/artigo06.htm), consulta em 15-07-2011).

LORENA, L., 2003, *Análise Espacial de Redes com Aplicações em Sistemas de Informações Geográficas*, (ABEPRO) (URL.: [www.lac.inpe.br/~lorena/producao/Analiseredes.pdf](http://www.lac.inpe.br/~lorena/producao/Analiseredes.pdf), consulta em 15-07-2011).

KEENAN, P., 2008, Modelling vehicle routing in GIS. *Operational Research Int J* 8, 201–218 (Springer-Verlag).

MILLER, J., Shaw, L., 2001, GIS-T Data Models. In *Geographic Information Systems for Transportation: Principles and Applications* (Oxford University Press) (URL: [www.gisvisionmag.com/Book/miller\\_shaw.pdf](http://www.gisvisionmag.com/Book/miller_shaw.pdf), consulta em 3-03-2010).

NETO, F., JUNIOR, F., 2002, Logística Empresarial, *Colecção Empresarial Gestão* (URL: <http://www.fae.edu/publicacoes/pdf/empresarial/4.pdf>, consulta em 16-07-2011).

RODRIGUE, J-P *et al.*, 2009, *The Geography of Transport Systems*, Hofstra University, Department of Global Studies & Geography, (URL: <http://people.hofstra.edu/geotrans> em 8-11-2009).

SILVA, A., RODRIGUES, D., ROSE, A., 2003, An Assessment Method for GIS-T Software. In *International Conference on Computers in Urban Planning and Urban Management (CUPUM'03): proceedings* (URL: <http://repositorium.sdum.uminho.pt/bitstream/1822/2318/1/11c1.pdf>, consulta em 8-11-2009).

THILL, J., 2000, Geographic information systems for transportation in perspective. In *Transportation Research Part C*, 3-12.

TOTH, P., VIGO, D. (Ed.s), 2002, *The Vehicle Routing Problem*, SIAM Monographs on Discrete Mathematics and Applications (URL: [books.google.pt/books?id=TeMgA5S74skC&printsec=frontcover&source=gbs\\_navlinks\\_s#v=onepage&q=&f=false](http://books.google.pt/books?id=TeMgA5S74skC&printsec=frontcover&source=gbs_navlinks_s#v=onepage&q=&f=false), consulta em 16-07-2011).

VITASEK, K., 2010, *Dictionary of Terms*, (Council of Supply Chain Professionals Management), (<http://cscmp.org/digital/glossary/glossary.asp>, consulta em 4-08-2011).

WATERS, N.M., 1999. Transportation GIS: GIS-T, Chapter 59. In *Geographical Information Systems*, 2nd Edition, Abridged, Longley, P.A., Goodchild, M.F..

MAGUIRE, D.J. & Rhind, D.W. (eds.), 1993, (Wiley). 59, 827-844 (URL: [ggs.gmu.edu/People/Waters/WatersGIS-T.pdf](http://ggs.gmu.edu/People/Waters/WatersGIS-T.pdf), consulta em 8-11-2009).

WIKIPEDIA, 2011, Página da Enciclopédia Livre (URL: <http://pt.wikipedia.org/wiki/>, consulta em 15-07-2011)

## ANEXOS

### Anexo 1 – *Scripts* em Python

```
# -----  
# Nome: rede2_pedidos.py  
# Descricao: script em linguagem python2.6, que sujeita os registos do ficheiro  
# pedidos.xlsx, a uma análise do tipo route usando a rede2 e devolve os nós  
# dos caminhos obtidos no ficheiro resultado_pedidos_rede2.dbf  
# -----  
  
# Importa Modulo Arcpy  
import arcpy  
  
#Verifica Licencas  
arcpy.checkoutextension("network")  
  
# Variaveis Locais:  
folha1_ = "c:\\sigitre\\pedidos.xlsx\\pedidos_xls$"  
eixos_tr_dgsp_nd = "c:\\sigitre\\rede2.gdb\\eixos_tr_dgsp\\eixos_tr_dgsp_nd"  
folha1__layer = "folha1$_layer"  
route_2__4_ = "route"  
route_2 = "route"  
out_shp = "c:\\sigitre\\outshp_ped.shp"  
eps_para_locate_locatefeatur5_dbf = "c:\\sigitre\\resultado_pedidos_rede2.dbf"  
output_event_table_properties = "rid point meas"  
eps_para_locate_shp = "c:\\sigitre\\rede2.gdb\\eixos_tr_dgsp\\eps_shp"  
  
# Metodo: Make Route Layer  
arcpy.makeroutelayer_na(eixos_tr_dgsp_nd, "route", "dist", "use_input_order",  
"preserve_both", "no_timewindows", "", "no_uturns", "oneway", "no_hierarchy", "",  
"true_lines_with_measures", "")
```

# Metodo: Make Xy Event Layer

```
arcpy.makexyeventlayer_management(folha1_, "coordx", "cordy", folha1__layer,
"geogcs['gcs_wgs_1984',datum['d_wgs_1984',spheroid['wgs_1984',6378137.0,298.2572
23563]],primem['greenwich',0.0],unit['degree',0.0174532925199433]];-400 -400
10000000000;-100000 10000;-100000 10000;8,98315284119521e-
09;0,001;0,001;ishighprecision", "")
```

# Metodo : Add Locations

```
arcpy.addlocations_na(route_2, "stops", folha1__layer, "name # #;routename file_id
#;timewindowstart # #;timewindowend # #;curbapproach # 0;attr_time # 0;attr_dist #
0", "50000 meters", "", "eixo_sul_shp none;eixo_norte_shp none;eixo_eps_norte_shp
none;eixos_epslx_alc_mon_shp none;eixos_gisp_lxmonalc_shp none;eixos_trib_lx_shp
none;eixos_centro_shp none;eixos_trib_aveiro_shp none;eps_shp
shape;eixos_tr_dgsp_nd_junctions shape", "match_to_closest", "append", "no_snap", "5
meters", "include", "eixo_sul_shp #;eixo_norte_shp #;eixo_eps_norte_shp
#;eixos_epslx_alc_mon_shp #;eixos_gisp_lxmonalc_shp #;eixos_trib_lx_shp
#;eixos_centro_shp #;eixos_trib_aveiro_shp #;eps_shp #;eixos_tr_dgsp_nd_junctions
#")
```

# Metodo : Solve

```
arcpy.solve_na(route_2__4_, "skip", "terminate")
```

# Metodo : Copy Features

```
arcpy.copyfeatures_management("route\\routes", out_shp, "", "0", "0", "0")
```

# Metodo : Locate Features Along Routes

```
arcpy.locatefeaturesalongroutes_lr(eps_para_locate_shp, out_shp, "name", "0
decimaldegrees", eps_para_locate_locatefeatur5_dbf, output_event_table_properties,
"all", "no_distance", "zero", "fields")
```

# Metodo : Delete

```

arcpy.delete_management(out_shp, "shapefile")

# -----
# Nome: rede3.py
# Descricao: script em linguagem python2.6, que sujeita os registos do ficheiro
# casos_escolhidos.xlsx, a uma análise do tipo route usando a rede3 e devolve
# os nós dos caminhos obtidos no ficheiro resultado_casos_rede3.dbf
# -----

# Importa Modulo Arcpy
import arcpy

# Verifica Licencas
arcpy.checkoutextension("network")

# Variaveis Locais:
pedidos = "c:\\sigitre\\casos_escolhidos.xlsx\\folha1$"
rede = "c:\\sigitre\\rede3.gdb\\eixos_tr_dgsp\\eixos_tr_dgsp_nd"
folha1__layer = "folha1$_layer"
route_2 = "route"
route_2__2_ = "route"
out_shp = "c:\\sigitre\\outshp3.shp"
eps_para_locate_locatefeatur5_dbf = "c:\\sigitre\\resultado_casos_rede3.dbf"
output_event_table_properties = "rid point meas"
eps_para_locate_shp = "c:\\sigitre\\rede3.gdb\\eixos_tr_dgsp\\eps_shp"

# Metodo : Make Route Layer
arcpy.makeroutelayer_na(rede, "route", "dist", "use_input_order", "preserve_both",
"no_timewindows", "", "allow_uturns", "oneway", "no_hierarchy", "",
"true_lines_with_measures", "")

```



# Metodo : Make Xy Event Layer

```
arcpy.makexyeventlayer_management(pedidos, "coordx", "cordy", folha1__layer,
"geogcs['gcs_wgs_1984',datum['d_wgs_1984',spheroid['wgs_1984',6378137.0,298.2572
23563]],primem['greenwich',0.0],unit['degree',0.0174532925199433]];-400 -400
1000000000;-100000 10000;-100000 10000;8,98315284119521e-
09;0,001;0,001;ishighprecision", "")
```

# Metodo : Add Locations

```
arcpy.addlocations_na(route_2, "stops", folha1__layer, "name # #;routename file_id
#;timewindowstart # #;timewindowend # #;curbapproach # 0;attr_time # 0;attr_dist #
0", "50000 meters", "", "eixo_sul_shp none;eixo_norte_shp none;eixo_eps_norte_shp
none;eixos_epslx_alc_mon_shp none;eixos_gisp_lxmonalc_shp none;eixos_centro_shp
none;eixos_trib_aveiro_shp none;eixos_lig_gisp_shp none;eixos_trib_lx_shp none;gisp
shape;eps_lig_gisp shape;eps_outros shape;eixos_tr_dgsp_nd_junctions shape",
"match_to_closest", "append", "no_snap", "5 meters", "include", "eixo_sul_shp
#;eixo_norte_shp #;eixo_eps_norte_shp #;eixos_epslx_alc_mon_shp
#;eixos_gisp_lxmonalc_shp #;eixos_centro_shp #;eixos_trib_aveiro_shp
#;eixos_lig_gisp_shp #;eixos_trib_lx_shp #;gisp #;eps_lig_gisp #;eps_outros
#;eixos_tr_dgsp_nd_junctions #")
```

# Metodo : Solve

```
arcpy.solve_na(route_2__2_, "skip", "terminate")
```

# Metodo : Copy Features

```
arcpy.copyfeatures_management("route\\routes", out_shp, "", "0", "0", "0")
```

# metodo : locate features along routes...

```
arcpy.locatefeaturesalongroutes_lr(eps_para_locate_shp, out_shp, "name", "0
decimaldegrees", eps_para_locate_locatefeatur5_dbf, output_event_table_properties,
"all", "no_distance", "zero", "fields")
```

# Metodo: Delete

```

arcpy.delete_management(out_shp, "shapefile")

# -----
# Nome: vrp_completo_gisp.py
# Descricao: script em linguagem python2.5, que faz uma análise
# do tipo vehicle routing problem usando a rede pedidos_vrp_nd2
# -----

# Importa Modulos
import sys, string, os, arcgisscripting

# Criacao do Objecto Geoprocessor
gp = arcgisscripting.create()

# verificacao de licencas
gp.checkoutextension("network")

# Carregamento de Toolboxes
gp.addtoolbox("c:/program files/arcgis/arctoolbox/toolboxes/network analyst tools.tbx")
gp.addtoolbox("c:/program files/arcgis/arctoolbox/toolboxes/data management
tools.tbx")

# Variaveis Locais
vrp_pyton = "vrp_pyton"
vrp_pyton__3_ = "vrp_pyton"
pedidos_vrp_nd2 =
"c:\\users\\public\\documents\\mapas_tese\\geodatabase_para_vrp.gdb\\pedidos_vrp\\pe
dididos_vrp_nd2"
poucos__layer = "poucos$_layer"
poucos_ =
"c:\\users\\public\\documents\\mapas_tese\\orders_vrp_poucos.xlsx\\poucos$"
vrp_pyton__2_ = "vrp_pyton"

```

```

depop_para_vrp__layer = "depop_para_vrp$_layer"
depop_para_vrp_                                     =
"c:\\users\\public\\documents\\mapas_tese\\depop_para_vrp_xlsx.xlsx\\depop_para_vrp
$"
vrp_pyton__4_ = "vrp_pyton"
route_vrp_gisp_shp = "c:\\users\\public\\documents\\mapas_tese\\route_vrp_gisp.shp"
vrp_pyton__5_ = "vrp_pyton"
od_poucos_dbf = "c:\\users\\public\\documents\\mapas_tese\\od_poucos.dbf"
network_analyst_layer = "vrp_pyton"
orders_output="c:\\users\\public\\documents\\mapas_tese\\orders_output_gisp.shp"

# Metodo: Make Vehicle Routing Problem Layer
gp.makevehicleroutingproblemlayer_na(pedidos_vrp_nd2,      "vrp_pyton",      "time",
"lenght", "minutes", "miles", "", "", "medium", "medium", "allow_urns", "",
"no_hierarchy", "", "true_lines_with_measures")

# Metodo: Make Xy Event Layer
gp.makexyeventlayer_management(poucos_,      "colo_coordx",      "colo_coordy",
poucos__layer, "")

# Metodo : Add Locations
gp.addlocations_na(vrp_pyton, "orders", poucos__layer, "name nome #;assignmentrule
# 3;curbapproach # 0;description # #;timewindowend1 trunc(pefi_data_prev_saida)
#;timewindowstart2 # #;servicetime # #", "5000 meters", "", "norte_vrp shape;sul_vrp
shape;vrp_eps shape;pedidos_vrp_nd2_junctions none", "match_to_closest", "append",
"no_snap", "5 meters")

# Metodo : Make Xy Event Layer
gp.makexyeventlayer_management(depop_para_vrp_,      "depotx",      "depoty",
depop_para_vrp__layer, "")

# Metodo : Add Locations

```

```
gp.addlocations_na(vrp_pyton__3_, "depots", depop_para_vrp_layer, "name name
#;curbapproach # 0", "5000 meters", "", "norte_vrp shape;sul_vrp shape;vrp_eps
shape;pedidos_vrp_nd2_junctions none", "match_to_closest", "append", "no_snap", "5
meters")
```

# Metodo: Add Locations

```
gp.addlocations_na(vrp_pyton__2_, "routes", route_vrp_gisp_shp, "name name
#;description # #;startdepotname startdepot #;enddepotname enddepotna
#;startdepotservicetime # #;enddepotservicetime # #;earlieststarttime earliestst
08:00:00;lateststarttime lateststar 10:00:00;capacities capacities #;fixedcost fixedcost
#;costperunittime # 1;costperunitdistance # #;overtimestarttime # #;costperunitovertime
# #;maxordercount # 30;maxtotaltime # #;maxtotaltraveltime # #;maxtotaldistance #
#;specialtynames # #;assignmentrule # 1", "5000 meters", "", "norte_vrp shape;sul_vrp
shape;vrp_eps shape;pedidos_vrp_nd2_junctions none", "match_to_closest", "append",
"no_snap", "5 meters")
```

# Metodo: Add Locations

```
gp.addlocations_na(vrp_pyton__4_, "order pairs", od_poucos_dbf, "maxtransittime
maxtransit #;firstordername firstorder #;secondordername secondorde #", "5000
meters", "", "norte_vrp shape;sul_vrp shape;vrp_eps shape;pedidos_vrp_nd2_junctions
none", "match_to_closest", "append", "no_snap", "5 meters")
```

# metodo: solve

```
gp.solve_na(vrp_pyton__5_, "halt")
```

# metodo: copy features

```
gp.copyfeatures_management("vrp_pyton\\orders", orders_output,"","0","0","0")
```

## **Anexo 2 – Tabelas e Procedimentos Oracle**

### **Criação Tabela Tfropini**

```
create table tfropini (  
    pini_id      varchar2 (5) not null,  
    pini_n_recl  varchar2 (15) not null,  
    pini_ep      varchar2 (5) not null,  
    pini_tipo_local varchar2 (3) not null,  
    pini_cod_local varchar2 (5) not null,  
    pini_data_prev date      not null,  
    primary key ( pini_id ) );
```

### **Criação Tabela Tfrocolo**

```
create table tfrocolo (  
    colo_tipo_local varchar2 (3) not null,  
    colo_cod_local  varchar2 (5) not null,  
    colo_coordx     number (14,7),  
    colo_coordy     number (14,7),  
    primary key ( colo_tipo_local, colo_cod_local ));
```

### **Criação View Vfropedu**

```
create or replace view vfropedu ( pedu_id,  
    pedu_tipo_local, pedu_cod_local, pedu_coordx, pedu_coordy) as select  
    to_number(pini_id),colo_tipo_local,pini_ep,colo_coordx,colo_coordy  
    from tfropini,tfrocolo  
    where pini_ep=colo_cod_local(+)  
    and colo_tipo_local=1  
    union  
    select  
    to_number(pini_id),pini_tipo_local,pini_cod_local,colo_coordx,colo_coordy  
    from tfropini, tfrocolo  
    where pini_cod_local=colo_cod_local(+)
```

and pini\_tipo\_local=colo\_tipo\_local(+)  
order by 1

### **Criação Tabela Tfropepa**

```
create table tfropepa (  
    pepa_pini      varchar2 (15) not null,  
    pepa_meas      varchar2 (16) not null,  
    pepa_tipo_local varchar2 (15),  
    pepa_cod_local varchar2 (15) not null,  
    pepa_eixo      varchar2 (15),  
    primary key ( pepa_pini, pepa_meas ) ) ;
```

### **Criação Tabela Tfropefi**

```
create table tfropefi (  
    pefi_tipo_tran  varchar2 (1),  
    pefi_cod_tran   varchar2 (15),  
    pefi_nome       varchar2 (50),  
    pefi_data_concessao date,  
    pefi_cod_saida  number (2),  
    pefi_data_prev_saida date,  
    pefi_cod_no_estab number (5),  
    pefi_tipo_no_estab varchar2 (3),  
    pefi_cod_no_origem number (5),  
    pefi_tipo_no_origem varchar2 (3),  
    pefi_cod_no_destino number (5),  
    pefi_tipo_no_destino varchar2 (3),  
    pefi_id_pini     varchar2 (5) not null,  
    pefi_id_order    varchar2 (3) not null,  
    primary key ( pefi_id_pini, pefi_id_order ) ) ;
```

### **Procedimento De\_Tfropepa\_Para\_Tfropefi**

create or replace procedure de\_tfropepa\_para\_tfropefi as

cursor cpepa is

select pini\_n\_recl, trunc(pini\_data\_prev-1) data, esta\_tran,(select cod\_estab from  
tfroesta where esta\_id\_sig=acodlocal) acodlocal,bcodlocal, to\_number(pini\_id) p,  
pn,arn

from

(select \*

from

(select pepa\_pini apini, pepa\_meas amead, pepa\_cod\_local acodlocal, pepa\_eixo aeixo,  
row\_number() over (partition by pepa\_pini order by pepa\_meas) as arn

from tfropepa pe

where pepa\_meas like

(select min(pepa\_meas)

from tfropepa

where pe.pepa\_pini=pepa\_pini

and pe.pepa\_eixo=pepa\_eixo

group by pepa\_pini,pepa\_eixo)

or pepa\_meas like

(select max(pepa\_meas)

from tfropepa

where pe.pepa\_pini=pepa\_pini

and pe.pepa\_eixo=pepa\_eixo

group by pepa\_pini,pepa\_eixo)

)a

,

(select pepa\_pini bpini, pepa\_meas bmeas, (select cod\_estab from tfroesta where  
esta\_id\_sig=pepa\_cod\_local) bcodlocal, pepa\_eixo beixo, row\_number() over (partition  
by pepa\_pini order by pepa\_meas) as brn,'l' pn

from tfropepa pe

where pepa\_meas like

```
(select min(pepa_meas)
from tfropepa
where pe.pepa_pini=pepa_pini
and pe.pepa_eixo=pepa_eixo
group by pepa_pini,pepa_eixo)
or pepa_meas like
(select max(pepa_meas)
from tfropepa
where pe.pepa_pini=pepa_pini
and pe.pepa_eixo=pepa_eixo
group by pepa_pini,pepa_eixo))b
where arn+1=brn
and apini=bpini
),tfroesta, tfropini
where esta_id_sig=acodlocal
and apini=pini_id
```

union

```
select pini_n_recl, pini_data_prev data, (select cod_estab from tfroesta where
esta_id_sig=acodlocal) trans, (select cod_estab from tfroesta where
esta_id_sig=acodlocal) acodlocal,bcodlocal, to_number(pini_id) p, pn,100
from
(select pepa_pini,pepa_cod_local acodlocal,pini_cod_local bcodlocal,pini_tipo_local pn
from tfropepa pe, tfropini
where pepa_meas like
(select max(pepa_meas)
from tfropepa
where pe.pepa_pini=pepa_pini
and pe.pepa_pini=pini_id
```



```

group by pepa_pini)),tfroesta, tfropini
where esta_id_sig=acodlocal
and pepa_pini=pini_id
order by p, arn;
data date;

rpepa cpepa%rowtype;

begin

for rpepa in cpepa loop
insert                                into                                tfropefi
(pefi_tipo_tran,pefi_cod_tran,pefi_data_prev_saida,pefi_cod_no_estab,
pefi_tipo_no_estab,pefi_cod_no_origem,    pefi_tipo_no_origem,pefi_cod_no_destino,
pefi_tipo_no_destino, pefi_id_pini, pefi_id_order)
values('r',rpepa.pini_n_recl,    rpepa.data,    rpepa.esta_tran,    '1',rpepa.acodlocal,
'1',rpepa.bcodlocal,rpepa.pn, rpepa.p, rpepa.arn);
end loop;
commit;
end de_tfropepa_para_tfropefi;

```

### **Procedimento Tratar\_Pedidos\_Partidos**

```

create or replace procedure tratar_pedidos_partidos as

cursor cpepa is
select pini_n_recl, esta_tran,acodlocal,bcodlocal
from
(
select *
from
(select pepa_pini apini, pepa_meas ameads, pepa_cod_local acodlocal, pepa_eixo aeixo,

```

```

rownum arn
from tfropepa pe
where pepa_meas like
(select min(pepa_meas)
from tfropepa
where pe.pepa_pini=pepa_pini
and pe.pepa_eixo=pepa_eixo
group by pepa_pini,pepa_eixo)
or pepa_meas like
(select max(pepa_meas)
from tfropepa
where pe.pepa_pini=pepa_pini
and pe.pepa_eixo=pepa_eixo
group by pepa_pini,pepa_eixo))a
,
(select pepa_pini bpini, pepa_meas bmeas, pepa_cod_local bcodlocal, pepa_eixo beixo,
rownum brn
from tfropepa pe
where pepa_meas like
(select min(pepa_meas)
from tfropepa
where pe.pepa_pini=pepa_pini
and pe.pepa_eixo=pepa_eixo
group by pepa_pini,pepa_eixo)
or pepa_meas like
(select max(pepa_meas)
from tfropepa
where pe.pepa_pini=pepa_pini
and pe.pepa_eixo=pepa_eixo
group by pepa_pini,pepa_eixo))b

where arn+1=brn

```

```

and apini=bpini),tfroesta, tfropini
where cod_estab=acodlocal
and apini=pini_id;

rpepa cpepa%rowtype;

begin
for rpepa in cpepa loop
insert      into      tfropefi      (pefi_tipo_tran,pefi_cod_tran,pefi_cod_no_estab,
pefi_tipo_no_estab,pefi_cod_no_origem,      pefi_tipo_no_origem,pefi_cod_no_destino,
pefi_tipo_no_destino)
values('r',rpepa.pini_n_recl, rpepa.esta_tran, '1',rpepa.acodlocal, '1',rpepa.bcodlocal, '1');
end loop;
commit;
end tratar_pedidos_partidos;Procedimento Insere_Pedidos_Partidos

```

### **Procedimento Tratar\_Pedidos\_Partidos**

create or replace procedure insere\_pedidos\_partidos as

```

cursor cpedidospartidos is
select pefi_cod_no_estab
,pefi_tipo_no_estab
,pefi_data_prev_saida
,pefi_tipo_no_origem
,pefi_cod_no_origem
,pefi_tipo_no_destino
,pefi_cod_no_destino
,pefi_cod_tran
,pefi_data_concessao
,pefi_nome
,pefi_cod_saida

```

```

from tfropefi
order by pefi_data_prev_saida desc;

begin
for rpedidospartidos in cpedidospartidos loop
fropktrajecto.importa_sip(
rpedidospartidos.pefi_tipo_no_estab,
rpedidospartidos.pefi_cod_no_estab,
rpedidospartidos.pefi_cod_tran,
rpedidospartidos.pefi_data_concessao,
rpedidospartidos.pefi_cod_saida,
sysdate,
rpedidospartidos.pefi_cod_no_origem,
rpedidospartidos.pefi_tipo_no_origem,
rpedidospartidos.pefi_cod_no_destino,
rpedidospartidos.pefi_tipo_no_destino,
rpedidospartidos.pefi_nome,
");

end loop;
end insere_pedidos_partidos;

```